Analysis of Random Sleep Scheme for Wireless Sensor Networks

Jianming Liu*, Xiaohong Jiang, Susumu Horiguchi
Graduate School of Information Science,
Tohoku Univeristy, Japan
E-mail: jmliu@guet.edu.cn, {jiang, susumu}@ecei.tohoku.ac.jp
*Corresponding author is also with GuiLin University of Electronic Technology.

Tony, Tong Lee
Department of Information Engineering,
Chinese University of Hong Kong, HKSAR, P. R. China
E-mail: ttlee@ie.cuhk.edu.hk

Abstract: Energy conserving design is the research focus to prolong the lifetime of wireless sensor networks. A simple and effective way to save energy is to place sensor nodes in sleep mode periodically. However, sleep mode corresponds to low power consumption as well as to reduced network capacity and increased latency. This paper develops an analytical framework to study the interaction between random sleep scheme and network performance. Our framework consists of the queueing model for sensor node and performance model for the whole network. We derived the network throughput, power consumption and packet delivery delay. Our analytical models shed light on the guidelines to design random sleep scheme and enable us to explore the trade-offs existing between sensor sleep/active dynamics and those performance measures.

Keywords: Wireless sensor network, Random sleep scheme, Power consumption, Throughput, Packet delivery delay

Reference to this paper should be made as follows: Jianming Liu, Xiaohong Jiang, Susumu Horiguchi and Tony-Tong Lee (2009) 'Analysis of random sleep scheme for wireless sensor networks', *Int. J. Sensor Networks*, Vol. x, Nos. x/x/x, pp.xx–xx.

Biographical notes: Dr.Jianming Liu received the B.Eng, B.Business Administration, M.Eng degrees from the University of Science and Technology of China, in 1999, 2002, respectively and Ph.D degree in Information Engineering from the Chinese University of Hong Kong in 2006. Currently, he is an associate professor of GuiLin University of Electronic Technology, GuangXi, P. R. China. Dr. Liu is also a GCOE research fellow of Tohoku University, Sendai, Japan.

Prof.Xiaohong Jiang received his B.S., M.S. and Ph.D degrees in 1989, 1992, and 1999 respectively, all from Xidian University, Xi'an, China. He is currently an Associate Professor in the Department of Computer Science, Tohoku University, Japan.

Prof.Susumu Horiguchi received the B.Eng, M.Eng and Ph.D degrees from Tohoku University in 1976, 1978 and 1981 respectively. He is currently a Full Professor in the Graduate School of Information Sciences, Tohoku University.

Prof.Tony-Tong Lee is a Professor of Information Engineering with Chinese University of Hong Kong. Dr. Lee is now serving as an Editor of the IEEE TRANSACTIONS ON COMMUNICATIONS. He also received the Leonard G. Abraham Prize Paper Award from the IEEE Communication Society in 1988 and Outstanding Paper Award from the IEICE of Japan in 1999. He is an IEEE Fellow since 2005.

1 Introduction

Recently, wireless sensor networks emerged from a wide variety of applications and systems with vastly varying requirements and characteristics. There is no standard or generic architecture for wireless sensor network to date. A *de facto* definition of a wireless sensor network has been established as an ad hoc multi-hop network which consists of large number tiny homogeneous sensor nodes that are resource-constrained, mostly immobile and randomly deployed in the area of interest [1]. Due to limited transmission range of sensor node, the sensory data are delivered to a processing center, called sink node, via multi-hop communication. Thus, each sensor node plays the dual roles of being a data originator and a data router.

The performances of a sensor network are measured in terms of lifetime, capacity and packet delay, which are determined by the dynamic behavior of sensor nodes, the efficiency of MAC protocol [2]-[7], the routing protocol [8]-[10] for packet delivery, and the topology of the whole network. Other specific wireless network issues such as channel contentions and transmission errors will affect the network performance as well.

Lifetime is the key performance measure for wireless sensor network, because it is very difficult or impossible to replace the power supplies for sensor nodes after deployed. In order to prolong the network lifetime, one popular approach based on MAC-layer design is to place nodes in sleep mode periodically [2], [3], [6], [7], [17]. The sensor node works under low-power operational state [6, 11] while it is in sleep mode. The longer time the node stays in sleep mode, the more energy can be saved.

It is shown in [2, 12] that simply putting sensor nodes in sleep mode for a fixed time interval would cause the network to synchronize and may lead to deadlock. Therefore, randomizing the sleep interval was proposed in [12]-[14] to resolve the deadlock problem while conserving energy at the same time. However, the energy conservation is realized at the expenses of reduced network capacity, increased latency and deferred system response. Analytical results on the capacity of large static ad hoc network with identical nodes are presented in [14]. Few analytical models have been developed to explore the trade-off between sleep scheme (energy consumption) and network performance. In [15, 16], the Markovian model and fluid model are developed, respectively, to analyze the homogeneous static sensor networks with random sleep scheme.

In this paper, we analyze a random sleep scheme with the assumptions that sensor nodes are randomly entering the sleep mode, and each node can be independently characterized by two exponentially distributed operational periods: Active and Sleep. In active mode, the node is fully working and able to transmit/receive packets. While in sleep mode, the node will turn off its transceiver, but continuously keep sensing the surrounding environment.

Under the preceding assumptions, an analytical model is developed to explore the impact of system parameters, such as the node sleep/active dynamics, maximum available routes and the time scale of sleep-active cycle, on network performance. Specifically, we develop a queueing model for the sensor nodes, and name it as *the single server queue with server shutting down in a varying neighborhood* to derive the probability measures of sensor node in different operational states, such as the packet number distribution, node throughput and responsive property. Next, we develop a power consumption model, which takes the node sleep to active transition frequency into consideration, and acquire the network capacity and mean packet delay.

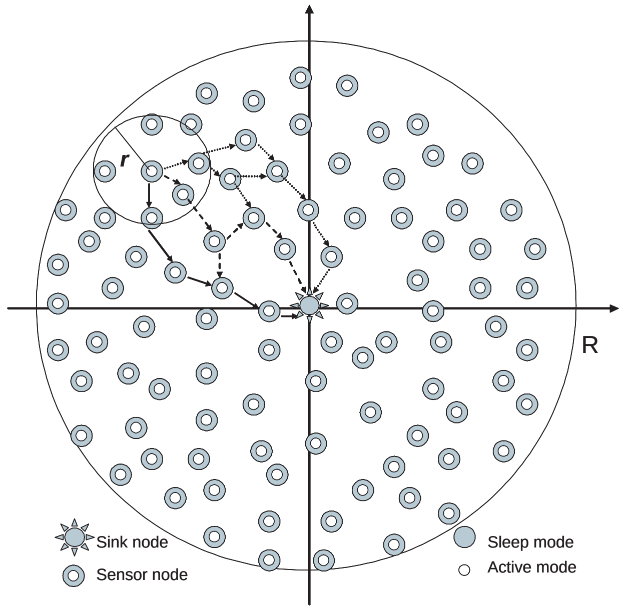

Figure 1: Wireless sensor network

Our results demonstrate that the sleep/active dynamics produces the predominant impact on average power consumption and mean packet delay. Properly choosing the time scale of sleep-active cycle is vital to the design of a power-efficient sensor network. Furthermore, our analysis also yields the estimate of maximum available routes that can be used to determine the proper routing table size.

The remainder of this paper is organized as follows: Section 2 introduces the operations of wireless sensor network, and the assumptions we made to carry out our analytical framework. Section 3 develops the sensor node queueing model and gets the packet number distribution in the sensor node. In Section 4, we derive the node throughput, mean packet number and analyze the node responsive property in sleep and active states. Section 5 presents network performance measures of interest. Section 6 explores the trade-off between random sleep scheme and network performance, and develops several guidelines to design power-efficient sensor network. Finally, the conclusion and direction of future work are summarized in Section 7.

2 Network Description and Assumptions

Assume that N static, identical sensor nodes each equipped with a full-duplex transceiver [19] [20] [21] are uniformly deployed over a disk with radius R. As depicted in Fig.1, the sink node located at the center of the disk collects the sensory data from all the nodes. Every node has an omni-directional antenna with maximum radio communication range r. Thus any two nodes within distance r can communicate directly, and there exists at least one path from any node to the sink. Although sensor nodes might be mobile, the nodes seldom move after deployed in many application scenarios [24], [25]. So we assume that the network topology is stationary and the cases that a sensor node is out of energy or malfunction are not considered.

2.1 Channel access and packet routing

Since wireless channel is shared among all nodes, an efficient Medium Access Control (MAC) is required. However, current protocols, such as IEEE 802.11 and 802.15.4, do not fit the characteristics of sensor networks, and cannot effectively support the sensor network applications. Although many customized MAC protocols [2]-[7] have been proposed, it remains an open question [4], [26], if a general, flexible MAC protocol exists that supports various applications while energy-efficient and offering acceptable network capability. In order to make it feasible to develop an analytical framework, which can still characterize the essential features of the whole sensor network, we assume that the wireless channel is error-free and the network is operated under an idealized contention-free MAC protocol, which means perfect synchronization during multi-hopping and no packet collisions during transmission.

Although several power efficient routing protocols have been proposed [8]-[10], in our analysis, we adopt the classical Dynamic Source Routing (DSR) algorithm [18] for its simplicity. Each node maintains a routing table that contains at most M routing paths (i.e. *next-hop nodes*) to the sink and each path is associated with its energy cost. We say M is the *maximum available route* of the node. The path with the lowest energy cost has the highest priority in the routing table. Upon transmitting data, the node polls the available next-hop nodes in turn according to their priorities determined by the associated energy cost. Thus, a node always dispatches its packets through the path available with highest energy efficiency. The routing algorithm also ensures that each node will not be overwhelmed by excessive relayed packets.

2.2 Random sleep scheme

With respect to energy consumption, the sensor node can be divided into three parts: *sensing unit*, *communication unit*, *data processing unit*(referred to Fig.2). Communication part is the major energy consumption part. If we actively shut down the communication part when necessary, remarkable energy can be saved [22]. In the random sleep scheme, we consider two distinguished modes: *active* and *sleep*. In *active* mode, the node is fully operational and can be in one of two states: *busy*(*transmit*/*receive*), and *idle*. While in *sleep* mode, the node only turns off its communication part, but remains its sensing unit active, i.e., the node will keep continuous sensing the surrounding environment in sleep mode. As depicted in Fig.3, each node will swing alternatively between *active*(A) and *sleep*(S) modes in a cycle manner. The times elapsed in active and sleep modes are exponentially distributed with mean $1/s$ and mean $1/a$, respectively.

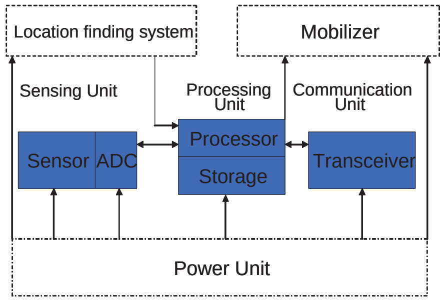

Figure 2: Components of sensor node

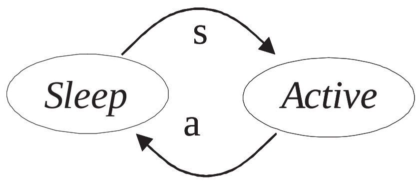

Figure 3: Sensor node random sleep scheme

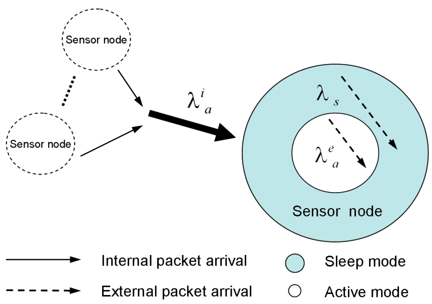

Figure 4: Two packet arrival processes in the sensor node

3 Sensor Node Queueing Analysis

3.1 Sensor node queueing model

Each node organizes the sensed data into short packets with variable size. This variation is due to the changes in compression (e.g., in MPEG video or perceptual audio codecs) or the amount of sensed data. Assume the packet size is exponentially distributed with mean size g Bytes. All the packets are stored in the buffer of sensor node. We assume that the system is stable and the buffer is modeled

Table 1: Notations in sensor node queueing model

W	Senor node transmission rate
g	Mean packet length
$\mu = W/8g$	Mean packet service rate
λ_a^e	Packet *generating* rate in active mode
λ_a^i	Arrival rate of *relayed* packets
$\lambda_a = \lambda_a^i + \lambda_a^e$	Packet arrival rate in *active* mode
λ_s	Packet generating rate in *sleep* mode
M	Maximum available route

as a single server FIFO queue. Also, by properly dimensioning the buffer so that the packet loss due to overflow is negligible, we can assume infinite buffer capacity [15] [16], which is the key to obtain close-form formulas of the node performance measures. Infinite buffer assumption also implies that all packets will eventually arrive at the sink node, which makes it feasible for us to analytically evaluate the interaction of network delay and energy consumption.

As each node is equipped with a full-duplex transceiver, it can receive and transmit packet simultaneously. Assume the transmission rate for each node is $W bps$.

In active mode, as depicted in Fig.4, the node senses the surrounding environment and generates packets according to a Poisson process with rate λ_a^e, and also relays the packets, dispatched by other nodes, arriving as a Poisson process with rate λ_a^i, where the superscript e means *external* and the superscript i means *internal*. So the packet arrival stream in active mode is a superposition of two Poisson process. Then, for active mode, we define the superposition Poisson process with rate λ_a, where

$$\lambda_a = \lambda_a^e + \lambda_a^i.$$

While in sleep mode, the node keeps sensing the surrounding environment and generates packets according to a Poisson process with rate λ_s but stop relaying for other nodes. Define

λ_a^e, λ_s:

External packet arrival, i.e., those packets are generated because of the node sensing the surrounding environment which is external with respect to the sensor network.

λ_a^i:

Internal packet arrival, i.e., the reference node relays packets for other nodes within the network.

The value of λ_a^e and λ_s are set beforehand when deploying the network, while λ_a^i is determined by the network scenario.

The mean packet service rate is

$$\mu = \frac{W}{8g}.$$

Observing under random sleep scheme, the whole network is essentially a collection of independent *active/sleep* (ON/OFF) processes, characterized by the distribution of the *active/sleep* periods. As depicted in Fig.5, from the viewpoint of a reference node, all its next-hop nodes behave like independent ON/OFF processes.

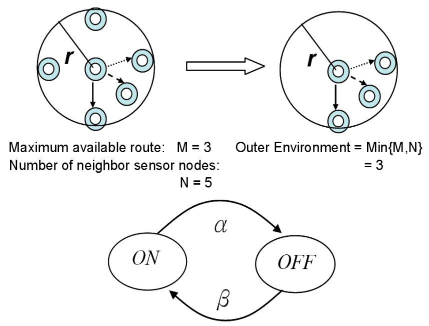

Figure 5: The *neighborhood* of sensor node

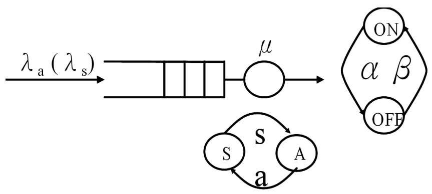

Figure 6: Sensor node queueing model

The *neighborhood* of the reference node is composed by the joint dynamics of all its next-hop nodes. For example in Fig.5, although the reference node has 5 neighbor nodes, if we limit the maximum available routes $M = 3$, the reference node can only have 3 next-hop nodes. If all the next-hop nodes are in sleep mode, the reference node can't forward packets to its next-hop nodes, we say its neighborhood is OFF. If there exists at least one next-hop node is active, the reference node can forward its packets, we say its neighborhood is ON. We use a continuous time Markov chain to model the varying of its neighborhood, as depicted in Fig.5.

The states of the reference node and its neighborhood can be adequately represented by the process

$$X(t) = \{y(t), m(t), n(t)\}$$

with the state space $\{(y, m, n)|y \in \{A, S\}, 0 \le m \le \infty, n \in \{0, 1\}\}$, where

$y(t) = A$: node in *Active* mode at time t
$y(t) = S$: node in *Sleep* mode
$m(t)$: number of packet(s) in the node at time t
$n(t) = 0$: neighborhood is ON at time t
$n(t) = 1$: neighborhood is OFF

Note that $y(t)$ denotes the random sleep scheme, governed by the Markov chain in Fig.3, while $n(t)$ denotes the dynamics of the node's neighborhood, governed by the Markov chain in Fig.5. The variations of $y(t)$ and $n(t)$ are corresponding to the vertical transitions in Fig.7.

The parallel transitions in Fig.7 are due to packet arrivals (λ_a or λ_s) and packet transmissions (μ). Notice that packet transmission happens only when the node is Active and its neighborhood is ON.

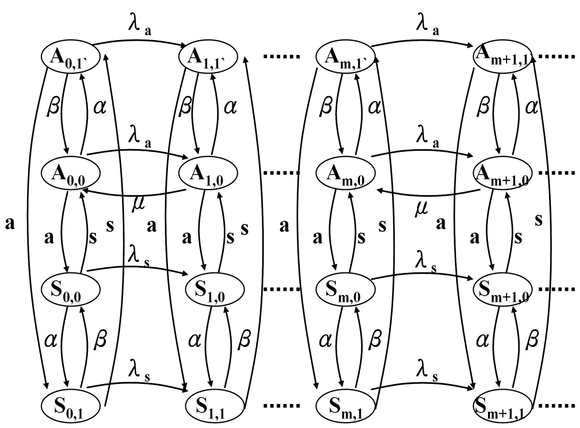

Figure 7: Transition diagram of sensor queueing model

For example, $A_{0,0}$ means node is Active, no packet in the node and its neighborhood is ON. If a packet arrived (with rate λ_a), $A_{0,0} \rightarrow A_{1,0}$; in state $A_{1,0}$, a packet gets transmitted results in $A_{1,0} \rightarrow A_{0,0}$. While $S_{0,0}$ means node is Sleep, no packet in the node and its neighborhood is ON, from which three possible transitions may happen:

1) a packet arrived (with rate λ_s)$\rightarrow S_{1,0}$
2) neighborhood becoming $OFF \rightarrow S_{0,1}$
3) node becoming Active $\rightarrow A_{0,0}$

Obviously, $X(t)$ is an Markov Process, and we can formulate the node behavior to a queueing model. We name it as *the single server queue with server shutting down in a varying neighborhood*, as depicted in Fig.6, where "*server shutting down*" means the node turns off its transceiver and enters sleep mode, while "*varying neighborhood*" means the variation of the joint status of next-hop nodes.

The neighborhood parameters α, β need to be estimated based on the network scenario and we will describe it in details in **Appendix A**. For the unknown rate λ_a^i of the relayed packets, we will derive it in details in the network model (**Appendix B**).

Define

$p^a_{m,n}$:

the probability with which the reference sensor node is *active*, m packets in the buffer and its neighborhood in state n.

$p^s_{m,n}$:

the probability with which the reference sensor node is *sleep*, m packets in the buffer and its neighborhood in state n.

For this queueing system, we can obtain the transition diagram like Fig.7 and the following Chapman-Kolmogorov equations (1), (2) and steady local balance equations (3),

$$\begin{cases} (\alpha+\beta+\lambda_a)p^a_{0,1} = \alpha p^a_{0,0} + s p^s_{0,1} \\ (s+\alpha+\lambda_s)p^s_{0,0} = a p^a_{0,0} + \beta p^s_{0,1} \\ (s+\beta+\lambda_s)p^s_{0,1} = a p^a_{0,1} + \alpha p^s_{0,0}, \end{cases} \tag{1}$$

$$\begin{cases} (\lambda_a+\mu+\alpha+a)p^a_{m,0} = \\ \qquad \mu p^a_{m+1,0} + \lambda_a p^a_{m-1,0} + \beta p^a_{m,1} + s p^s_{m,0} \\ (\lambda_a+\beta+a)p^a_{m,1} = \lambda_a p^a_{m-1,1} + \alpha p^a_{m,0} + s p^s_{m,1} \\ (\lambda_s+\alpha+s)p^s_{m,0} = \lambda_s p^s_{m-1,0} + \beta p^s_{m,1} + a p^a_{m,0} \\ (\lambda_s+\beta+s)p^s_{m,1} = \lambda_s p^s_{m-1,1} + \alpha p^s_{m,0} + a p^a_{m,1}, \end{cases} \tag{2}$$

$$\begin{cases} \mu p^a_{m+1,0} = \lambda_a p^a_{m,0} + \lambda_a p^a_{m,1} + \lambda_s p^s_{m,1} + \lambda_s p^s_{m,1} \\ (a+\beta+\lambda_a)p^a_{m+1,1} = \lambda_a p^a_{m,1} + \alpha p^a_{m+1,0} + s p^s_{m+1,1} \\ (s+\alpha+\lambda_s)p^s_{m+1,0} = \lambda_s p^s_{m,0} + \beta p^s_{m+1,1} + a p^a_{m+1,0} \\ (s+\beta+\lambda_s)p^s_{m+1,1} = \lambda_s p^s_{m,1} + \alpha p^s_{m+1,0} + a p^a_{m+1,1}. \end{cases} \tag{3}$$

In the following sections, first we will resolve the queueing model to get the packet number distribution, and then we will get the node throughput, mean packet number in sensor node and analyze the node responsive property.

3.2 Packet number distribution in sensor node

Our analysis starts from the initial probability $p^a_{0,0}$. We will derive the relationship between $p^a_{0,0}$ and the other state probabilities. And then we use the standard normalization condition to resolve the queueing model.

From the last two equations of (1), we can get the relationship between $\mathbf{P}^A_0$ and $\mathbf{P}^S_0$

$$\mathbf{P}^S_0 = \mathbf{D}\mathbf{P}^A_0, \tag{4}$$

where

$$\mathbf{P}^A_0 = \begin{bmatrix} p^a_{0,0} \\ \\ p^a_{0,1} \end{bmatrix}, \mathbf{P}^S_0 = \begin{bmatrix} p^s_{0,0} \\ \\ p^s_{0,1} \end{bmatrix},$$

$$\mathbf{D} = \begin{bmatrix} \frac{a(s+\beta+\lambda_s)}{(s+\alpha+\lambda_s)(s+\beta+\lambda_s)-\alpha\beta} & \frac{a\beta}{(s+\alpha+\beta)(s+\beta+\lambda_s)-\alpha\beta} \\ \\ \frac{a\alpha}{(s+\alpha+\lambda_s)(s+\beta+\lambda_s)-\alpha\beta} & \frac{a(s+\alpha+\lambda_s)}{(s+\alpha+\beta)(s+\beta+\lambda_s)-\alpha\beta} \end{bmatrix}.$$

Combining (4) with the first equation of (1) , we can get the relationship between $p^a_{0,0}$ and $p^a_{0,1}$ as following

$$p^a_{0,1} = \frac{\alpha + sD_{21}}{(\alpha+\beta+\lambda_a) - sD_{22}} p^a_{0,0}. \tag{5}$$

Define

$$\mathbf{K} = \begin{bmatrix} 1 \\ \\ \frac{\alpha + sD_{21}}{(\alpha+\beta+\lambda_a) - sD_{22}} \end{bmatrix},$$

we have

$$\mathbf{P}^A_0 = \mathbf{K} p^a_{0,0}. \tag{6}$$

To apply the normalization condition, now we will derive two recursive equations and these two equations can be used to calculate the packet number distribution. We can derive the first recursive equation from the last two equations of (2)

$$\mathbf{P}^S_{m+1} = \mathbf{C}\mathbf{P}^S_m + \mathbf{D}\mathbf{P}^A_{m+1} \;\; (m = 0, 1, 2...), \tag{7}$$

where

$$\mathbf{P}_m^A = \begin{bmatrix} p_{m,0}^a \\ \\ p_{m,1}^a \end{bmatrix}, \mathbf{P}_m^S = \begin{bmatrix} p_{m,0}^s \\ \\ p_{m,1}^s \end{bmatrix},$$

$$\mathbf{C} = \begin{bmatrix} \frac{\lambda_s(s+\beta+\lambda_s)}{(s+\alpha+\lambda_s)(s+\beta+\lambda_s)-\alpha\beta} & \frac{\beta\lambda_s}{(s+\alpha+\beta)(s+\beta+\lambda_s)-\alpha\beta} \\ \\ \frac{\alpha\lambda_s}{(s+\alpha+\lambda_s)(s+\beta+\lambda_s)-\alpha\beta} & \frac{\lambda_s(s+\alpha+\lambda_s)}{(s+\alpha+\beta)(s+\beta+\lambda_s)-\alpha\beta} \end{bmatrix}.$$

Deriving from (7) and the second equation of (2), we get

$$p_{m+1,1}^a = \frac{(\alpha+sD_{21})\lambda_a}{[(a+\beta+\lambda_a)-sD_{22}]\mu} p_{m,0}^a + \frac{(\alpha+sD_{21}+\mu)\lambda_a}{[(a+\beta+\lambda_a)-sD_{22}]\mu} p_{m,1}^a + \frac{(\alpha+sD_{21})\lambda_s+sC_{21}\mu}{[(a+\beta+\lambda_a)-sD_{22}]\mu} p_{m,0}^s + \frac{(\alpha+sD_{21})\lambda_s+sC_{22}\mu}{[(a+\beta+\lambda_a)-sD_{22}]\mu} p_{m,1}^s. \quad (8)$$

Combining (8) with the first equation of (2) yields the second recursive equation for deriving the packet number distribution

$$\mathbf{P}_{m+1}^A = \mathbf{G}\mathbf{P}_m^A + \mathbf{L}\mathbf{P}_m^S \ \ (m = 0, 1, 2...), \quad (9)$$

where

$$\mathbf{G} = \begin{bmatrix} \frac{\lambda_a}{\mu} & \frac{\lambda_a}{\mu} \\ \\ \frac{(\alpha+sD_{21})\lambda_a}{[(a+\beta+\lambda_a)-sD_{22}]\mu} & \frac{(\alpha+sD_{21}+\mu)\lambda_a}{[(a+\beta+\lambda_a)-sD_{22}]\mu} \end{bmatrix}$$

$$\mathbf{L} = \begin{bmatrix} \frac{\lambda_s}{\mu} & \frac{\lambda_s}{\mu} \\ \\ \frac{(\alpha+sD_{21})\lambda_s+sC_{21}\mu}{[(a+\beta+\lambda_a)-sD_{22}]\mu} & \frac{(\alpha+sD_{21})\lambda_s+sC_{22}\mu}{[(a+\beta+\lambda_a)-sD_{22}]\mu} \end{bmatrix}.$$

Combining the two recursive equations (7) and (9) yields

$$\begin{cases} \mathbf{P}_{m+1}^A = \mathbf{G}\mathbf{P}_m^A + \mathbf{L}\mathbf{P}_m^S \\ \\ \mathbf{P}_{m+1}^S = \mathbf{C}\mathbf{P}_m^S + \mathbf{D}\mathbf{P}_{m+1}^A \ \ (m = 0, 1, 2...). \end{cases} \quad (10)$$

Now we can resolve the queueing model from (10)

$$\begin{cases} \sum_{m=0}^{\infty} \mathbf{P}_{m+1}^A = \mathbf{G}\sum_{m=0}^{\infty} \mathbf{P}_m^A + \mathbf{L}\sum_{m=0}^{\infty} \mathbf{P}_m^S \\ \\ \sum_{m=0}^{\infty} \mathbf{P}_{m+1}^S = \mathbf{C}\sum_{m=0}^{\infty} \mathbf{P}_m^S + \mathbf{D}\sum_{m=0}^{\infty} \mathbf{P}_{m+1}^A, \end{cases} \quad (11)$$

and then (11) yields

$$\begin{cases} \mathbf{P}^A - \mathbf{P}_0^A = \mathbf{G}\mathbf{P}^A + \mathbf{L}\mathbf{P}^S \\ \\ \mathbf{P}^S - \mathbf{P}_0^S = \mathbf{C}\mathbf{P}^S + \mathbf{D}(\mathbf{P}^A - \mathbf{P}_0^A,) \end{cases} \quad (12)$$

where

$$\mathbf{P}^S = \sum_{m=0}^{\infty} \mathbf{P}_m^S, \quad \mathbf{P}^A = \sum_{m=0}^{\infty} \mathbf{P}_m^A.$$

The first equation of (12) can be written down as

$$\mathbf{P}^S = \mathbf{L}^{-1}[(\mathbf{I} - \mathbf{G})\mathbf{P}^A - \mathbf{P}_0^A]. \quad (13)$$

Inserting (13) into the second equation of (12) and combining the result with (4) and (6) yields

$$\begin{cases} \mathbf{P}^A = \Delta\mathbf{K}p_{0,0}^a \\ \\ \mathbf{P}^S = \mathbf{L}^{-1}[(\mathbf{I} - \mathbf{G})\Delta - \mathbf{I}]\mathbf{K}p_{0,0}^a, \end{cases} \quad (14)$$

where

$$\Delta = \Big[(\mathbf{I} - \mathbf{C})\mathbf{L}^{-1}(\mathbf{I} - \mathbf{G}) - \mathbf{D}\Big]^{-1}(\mathbf{I} - \mathbf{C})\mathbf{L}^{-1}.$$

Define $e = [1, 1]$, by using the normalization condition

$$e(\mathbf{P}^A + \mathbf{P}^S) = 1, \quad (15)$$

we can finally obtain the initial probability to resolve the sensor node queueing model

$$p_{0,0}^a = \frac{1}{X_1 + X_2}, \quad (16)$$

where

$$\begin{cases} X_1 = e\Delta\mathbf{K} \\ X_2 = e\mathbf{L}^{-1}[(\mathbf{I} - \mathbf{G})\Delta - \mathbf{I}]\mathbf{K}. \end{cases}$$

Now as depicted in the following procedure, we can get the packet number distribution in sensor node and then derive several node performance measures of interest.

1. Using normalization condition (16) to calculate the initial probability $p_{0,0}^a$;
2. Using (6) to determine $\mathbf{P}_0^A$;
3. Using (4) to determine $\mathbf{P}_0^S$;
4. Recursively calculate the packet number distribution from the following equation set

$$\begin{cases} \mathbf{P}_{m+1}^A = \mathbf{G}\mathbf{P}_m^A + \mathbf{L}\mathbf{P}_m^S \\ \\ \mathbf{P}_{m+1}^S = \mathbf{C}\mathbf{P}_m^S + \mathbf{D}\mathbf{P}_{m+1}^A \ \ (m = 0, 1, 2...) \end{cases}$$

4 Sensor Node Performance

4.1 Probabilities of different node states

In the above section, we have resolved the sensor node queueing model, now we can thoroughly get some probability measures to characterize the behavior of sensor nodes.

The node can operate in *Sleep* or *Active* mode: in active mode, the node may be idle or receiving/transmitting (i.e. forwarding) packets. To get the node throughput and mean packet number, we calculate the probabilities of different operational states first.

In random sleep scheme, the time duration in *active/sleep* modes are both exponentially distributed, it is easy to get the probability of sensor node in active state as

$$P_a = \frac{s}{s + a}, \quad (17)$$

and the probability in sleep mode is

$$P_s = \frac{a}{s+a}. \tag{18}$$

When the reference node is active and its neighborhood is ON, the node can forward packets to its next-hop nodes, so the probability of node actually forwarding its packets is

$$P_{af} = P_{a,0} - p^a_{0,0}. \tag{19}$$

In (19), $P_{a,0}$, the probability of node in active mode and its neighborhood in ON state, can be calculated from the following equation (20). In other words, $P_{a,0}$ is the probability with which sensor node can forward its packets. While $p^a_{0,0}$ is the probability with which sensor node is active, and its neighborhood is ON but no packet needs to be transmitted. That is to say, $p^a_{0,0}$ is the probability that the node can forward its packets but no packet in it. $p^a_{0,0}$ can be calculated from (16).

Define $e_0 = [1,0]$,$e_1 = [0,1]$, we have

$$P_{a,0} = e_0\mathbf{P}^A. \tag{20}$$

While the probability of sensor node in active mode and its neighborhood in OFF state is

$$P_{a,1} = e_1\mathbf{P}^A. \tag{21}$$

In (20),(21), $\mathbf{P}^A$ can be calculated from (14).

4.2 Node throughput and mean packet number

In the node queueing model, the average packet service rate is $\mu = \frac{W}{8g}$. For a sensor node, average packet throughput is

$$T = P_{af}\mu. \tag{22}$$

Mean packet number in the node is [see **Appendix C**]

$$\overline{M} = e^M\Omega^{-1}\Gamma, \tag{23}$$

in (23), we have

$$e^M = [1,1,1,1], \mathbf{\Omega} = \begin{bmatrix} (\alpha+a) & -\beta & -s & 0 \\ -\alpha & (\beta+a) & 0 & -s \\ -a & 0 & (\alpha+s) & -\beta \\ (\lambda_a-\mu) & \lambda_a & \lambda_s & \lambda_s \end{bmatrix}$$

$$\mathbf{\Gamma} = \begin{bmatrix} (\lambda_a P_{a,0} - \mu P_{af}) \\ \lambda_a P_{a,1} \\ \lambda_s P_{s,0} \\ -\mu P_{af} \end{bmatrix}, P_{s,0} = e_0\mathbf{P}^S.$$

4.3 Node responsive property

Sensor nodes entering sleep mode periodically will defer the network response and increase the latency. To evaluate the node responsive property in *sleep* and *active* mode, we make the following definitions:[Details refereed to **Appendix D**].

$\overline{W}_m$:mean waiting time conditioned on m packets in sensor node and the neighborhood is ON, the node is *active*;

$\overline{V}_m$:mean waiting time conditioned on m packets in sensor node and the neighborhood is OFF, the node is *active*;

$\overline{O}_m$:mean waiting time conditioned on m packets in sensor node and the neighborhood is ON, the node is *sleep*;

$\overline{H}_m$:mean waiting time conditioned on m packets in sensor node and the neighborhood is OFF, the node is *sleep*.

$\overline{W}_m, \overline{V}_m, \overline{O}_m, \overline{H}_m (m = 0,1,2...)$ can be recursively calculated using the following equations

$$\overline{W}_m = \frac{\mu}{\mu + a(1-c-d)}\overline{W}_{m-1} + \frac{1+\alpha h + a(dh+g)}{\mu+a(1-c-d)} \tag{24}$$
$$(m = 1,2...),$$

$$\begin{cases} \overline{V}_m = \overline{W}_m + h \\ \overline{O}_m = c\overline{W}_m + d\overline{V}_m + \frac{1}{s} \quad (m = 0,1,2...), \\ \overline{H}_m = k\overline{W}_m + f\overline{V}_m + \frac{1}{s} \end{cases} \tag{25}$$

and we have $\overline{W}_0 = 0$, which means when a packet arrives in *active* mode, no other packet waiting in the buffer and the neighborhood is ON, this packet needn't to wait and can be transmitted directly.

In equation (24) and (25), we have defined

$$h = \frac{s^2 + (\alpha+\beta)s + a(s+\alpha+\beta)}{s[(\alpha+\beta)(s+\alpha+\beta) - a(\alpha+s)]},$$

$$c = \frac{s+\beta}{s+\alpha+\beta}, \qquad d = \frac{\alpha}{s+\alpha+\beta},$$
$$k = \frac{\beta}{s+\alpha+\beta}, \qquad f = \frac{s+\alpha}{s+\alpha+\beta}.$$

From $\overline{W}_m, \overline{V}_m, \overline{O}_m, \overline{H}_m$ we can evaluate the responsive ,property of the sensor node as following:
i)With probability P_a, sensor node is active. When node is *active*, if a packet arrives, it will have to wait average time $\overline{D}^a_{node}$ to get service,

$$\overline{D}^a_{node} = \sum_{m=0}^{\infty} \frac{p^a_{m,0}\overline{W}_m + p^a_{m,1}\overline{V}_m}{P_a}. \tag{26}$$

ii)With probability P_s, sensor node is sleep. When node is *sleep*, if a packet arrives, it will have to wait average time $\overline{D}^s_{node}$ to get service,

$$\overline{D}^s_{node} = \sum_{m=0}^{\infty} \frac{p^s_{m,0}\overline{O}_m + p^s_{m,1}\overline{H}_m}{P_s}. \tag{27}$$

5 Network Capacity, Delay and Power Consumption

From the senor node model (Section 3.1) and network model (**Appendix B**), we can derive the network capacity, packet delivery delay and power consumption.

Define Λ_E^k as the *equivalent packet generating rate* at node k from the surrounding environment. Considering node k is active with probability P_a and is sleep with probability P_s, we have

$$\Lambda_E^k = P_a \lambda_a^{e,k} + P_s \lambda_s^k. \tag{28}$$

Notice that we assume the nodes have infinite buffer, so the packets never loss during traversing the network, and every packet will arrive at the sink node eventually. Then the whole network capacity C is the average packet arrival rate at the sink, which is simply the overall equivalent packet generating rate of all the sensor nodes

$$C = \sum_{k=1}^{N} \Lambda_E^k = \sum_{k=1}^{N} (P_a \lambda_a^{e,k} + P_s \lambda_s^k). \tag{29}$$

The mean packet delivery delay $\overline{D}_{net}$, which is the mean duration experienced by a packet from nodes to the sink. From Little's law [23] of the whole network, we can calculate $\overline{D}_{net}$ as

$$\overline{D}_{net} = \frac{\sum_{k=1}^{N} \overline{M}^k}{C}, \tag{30}$$

where $\overline{M}^k$ is the mean packet number in sensor node k and can be obtained from (23).

The power consumption of each node can be divided into three parts. The first part is the power consumption due to the node operational mode: Pw_s and Pw_a, where Pw_s and Pw_a are the power consumptions in *sleep* and *active* modes, respectively. The second part is the power required to transmit and receive packets: Pw_{tran}, Pw_{recv}, respectively. And the third part is the energy required during transition from sleep mode to active mode:E_{tr}. In Section 3.2 we have obtained P_s, P_a, P_{af} which are the probabilities of the node in *sleep*, *active* and packet *transmitting* states.

Note that the mean sleep duration $\overline{T}_{sleep}$ is $1/s$ and the mean active duration $\overline{T}_{active}$ is $1/a$. So the mean duration of the *sleep-active* cycle is

$$\overline{T}_{cycle} = \overline{T}_{sleep} + \overline{T}_{active}. \tag{31}$$

We know that in each sleep-active cycle there is only one transition from sleep mode to active mode and this transition will cost energy E_{tr}. So the *equivalent power* Pw_{tr} needed for this transition will be

$$Pw_{tr} = \frac{E_{tr}}{\overline{T}_{cycle}}. \tag{32}$$

Now the power consumption of a single sensor node is

$$Pw = P_s Pw_s + P_a Pw_a + P_{af} Pw_{tran} + P_a Pw_{recv} + Pw_{tr}. \tag{33}$$

Correspondingly, the power consumption of the whole network is

$$PW = \sum_{k=1}^{N} Pw^k. \tag{34}$$

And the average node power consumption is

$$\overline{W} = \frac{PW}{N}. \tag{35}$$

Table 2: Parameter setting in sensor network simulation

Network Radius: R	250 m
Node Communication Radius: r	25 m
Number of Nodes: N	400
Transmission Rate: W	1 Mbps
Mean Packet Length: g	80 *Bytes*
Power in Active mode: Pw_a	15 mW
Power in Sleep mode: Pw_s	3 mW
Transmission Power: Pw_{tran}	100 mW
Receiving Power: Pw_{recv}	20 mW
Energy for Sleep to Active Transition: E_{tr}	0.2 mJ

6 Network Performance under Random Sleep Scheme

In this section, we will explore the design space to shed light to the impact of design parameters on network performance. Those design parameters are traffic load (network capacity), random sleep scheme (i.e. sleep/active dynamics), frequency of sleep to active transition and maximum available routes. To evaluate the impact of traffic load to network performance, we first define a theoretical network load similar to that of [15] as

$$\rho = \frac{N(P_a \lambda_a^e + P_s \lambda_s)}{\mu} \tag{36}$$

Note that ρ represents the sum of traffic loads generated by all the nodes as if they were in isolation, only including parameters that are set before deploying the network.

For node sleep/active dynamics, we define the node *duty cycle* as the fraction of time the node is in sleep mode and we use a/s to characterize it.

$$DutyCycle = \frac{a}{s} \tag{37}$$

The parameter setting in our simulation is presented in Table 2. Without specification, we assume that all the sensor nodes generate packets with a same rate (i.e., all nodes have same λ_a^e and λ_s) and the number of maximum available routes is $M = 5$.

Fig.8∼Fig.10 present the trade-off between average power consumption and average delay with network traffic load $\rho = 0.9, 0.4, 0.1$ respectively. Analytical average power consumption and average delay can be obtained by (33)∼(35) and (30) respectively.

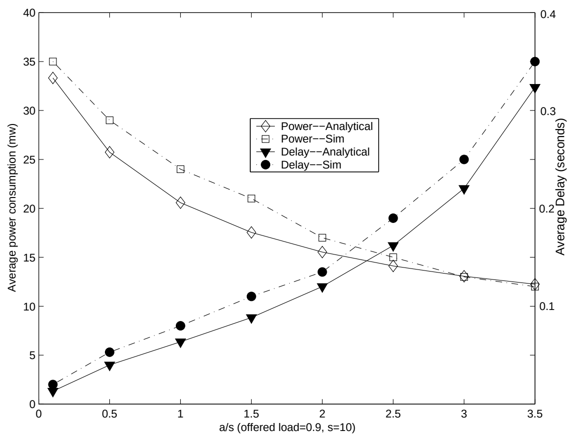

Figure 8: Trade-off between average power consumption and average delay (traffic load $\rho = 0.9, s = 10$)

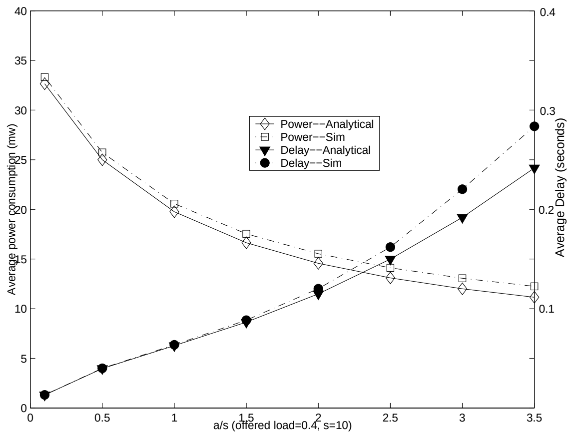

Figure 9: Trade-off between average power consumption and average delay (traffic load $\rho = 0.4, s = 10$)

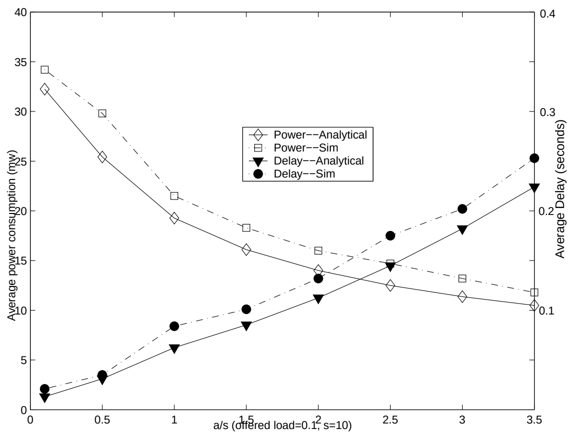

Figure 10: Trade-off between average power consumption and average delay (traffic load $\rho = 0.1, s = 10$)

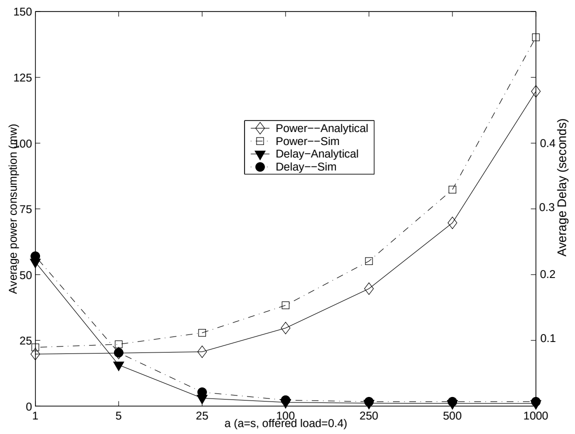

Figure 11: Trade-off between average power consumption and average delay under different transition frequency (traffic load $\rho = 0.4, a = s$)

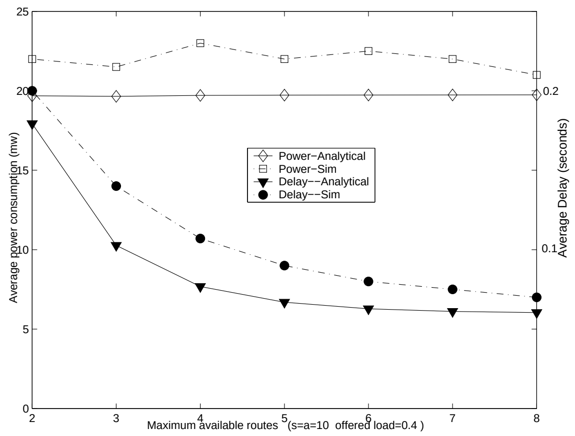

Figure 12: Trade-off between average power consumption and average delay under different maximum available route (traffic load $\rho = 0.4, a = s = 10$)

In our following analysis, we set $s = 10$, which means the mean sleep duration is $100ms$. We also fix the network traffic load and change the value of a. For example, if $a = 1$, $a/s = 0.1$, which means the mean active duration is $1000ms$, it is 10 times of the sleep duration. If a/s is small, sensor nodes will work in high *duty cycle* and the nodes will spend much time in active mode. We then obtain a small average packet delivery delay with the expense of very large average power consumption.

And when a/s becomes larger, sensor nodes will work in lower *duty cycle* and the nodes will spend less time in active mode, the average power consumption will be reduced and the penalty is that packets have to experience longer average delay.

From Fig.8∼Fig.10 we can see that average power consumption is not sensitive to network traffic load, because from the definition of ρ, we can see that even $\rho = 0.9$, traffic load of single node are very small, and then power consumption due to Recv/Tran is very small too. On the contrary, network traffic load has greater impact on the average packet delivery delay than on average power consumption. It is the sleep/active dynamics that is the key parameter to dominate the power consumption and delay.

Fig.11 presents an interesting trade-off between average power consumption and average delay with respect to the *transition frequency* with which the nodes transit from sleep to active mode. In our analysis, we set $a = s$ and change a from 1 to 1000, which means the mean duration of sleep-active cycle will change from $2000ms$ to $2ms$. The average power consumption increases dramatically from around $20mW$ to nearly $120mW$. This is because when the transition frequency increases, the equivalent power needed to transit from sleep to active mode will become bigger and bigger and dominate the overall node power consumption finally. From Fig.11 we know that properly choosing the time scale of sleep-active cycle is very important to design a power-efficient sensor network.

Fig.12 shows the impact of maximum available route M on the average power consumption and average packet delivery delay. We can see that when M is small (e.g., $M = 2$), the average delay is relative bigger and increasing M will reduce the delay effectively (e.g., M from 2 to 5); but continuously increasing M from 6 to 8 will not reduce the delay effectively. On the other hand, the maximum available route M has little impact on the average power consumption. Based on Fig.12, we can choose an optimal maximum available routes $M = 5$ to limit the routing table size while still guarantee the network performance.

7 Conclusion and Future Work

This paper analyzes the wireless sensor network whereby the nodes enter sleep mode randomly, independent of each other, and transmit their data to sink node via multi-hop communication.

We derived the node queueing model to analyze the random sleep scheme, and developed an analytical framework to model the network performance. For single sensor node, we derived its node throughput, mean packet number and analyzed its responsive property. For the whole network, we derived the power consumption, capacity and mean packet delay. We explored the trade-off among the above-mentioned performance measures and validated our analytical results through extensive simulations.

We conclude that node sleep/active dynamics is the key parameter which has greatest impact on the average power consumption and mean packet delivery delay. Properly choosing the time scale of sleep-active cycle is vital to design a power-efficient sensor network.

In our current research, we have assumed an idealized MAC protocol, so one of our future works is to extend the framework to incorporate with more realistic protocol with collisions and transmission errors. In a practical sensor network, the node buffer size is finite. Evaluating the impact of buffer size on the network performance is another objective of our future work. Finally, notice that the sensor nodes may be mobile and heterogeneous, how to integrate the node mobility and heterogeneity into an analytical framework is a challenging issue.

ACKNOWLEDGMENT

This work was supported in part by the GCOE project of Tohoku University, JSPS Grant-In-Aid of Scientific Research(C)19500050, the National Natural Science Foundation of China (60762002,60763004) and GuangXi province (0731024), GuiJiaoKeYan 2006(No.26, D200644), the State Key Laboratory for Novel Software Technology, Nanjing University, China.

REFERENCES

[1] K. Romer and F. Mattern,"The design space of wireless sensor networks," *IEEE Wireless Communications*, Vol. 11, Issue 6, Dec. 2004, pp: 54 - 61.

[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, June 2004.

[3] M. J. Miller and N. H. Vaidya, "A MAC protocol to reduce sensor network energy consumption using a wakeup radio," *IEEE Transactions on Mobile Computing*. Vol. 4, Issue 3, May-June 2005, pp: 228 - 242.

[4] Y. E. Sagduyu and A. Ephremides, "The problem of medium access control in wireless sensor networks," *IEEE Wireless Communications*, Vol. 11, Issue 6, Dec. 2004, pp: 44 - 53.

[5] Y. C. Tay, K. Jamieson and H. Balakrishnan, "Collision minimizing CSMA and its applications to wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, Vol. 22, Issue 6, Aug. 2004, pp: 1048 - 1057.

[6] S. Singh and C. S. Raghavendra, "PAMAS: power aware multi-access protocol with signaling for ad hoc networks", *ACM Computer Communication Review*, July 1998, pp: 5 - 26.

[7] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, "Topology management for sensor networks: exploiting latency and density," *Proceeding of ACM MobiHoc Conference*, June 2002.

[8] Jae-Hwan Chang and L. Tassiulas, "Maximum lifetime routing in wireless sensor networks," *IEEE/ACM Transactions on Networking*, Vol. 12, Issue 4, Aug. 2004, pp: 609 - 619.

[9] S. D. Muruganathan, D. C. F. Ma, R. I. Bhasin and A. O. Fapojuwo, "A centralized energy-efficient routing protocol for wireless sensor networks," *IEEE Communications Magazine*, Vol. 43, Issue 3, March 2005, pp: S8 - 13.

[10] R. Jain, A. Puri and R. Sengupta, "Geographical routing using partial information for wireless ad hoc networks," *IEEE Personal Communications*, Vol. 8, Issue 1, pp: 48 - 57.

[11] A. Sinha and A. P. Chandrakasan, "Dynamic power management in wireless sensor networks," *IEEE Design and Test of Computers Magazine*, Vol. 18, No. 2, March - April 2001, pp: 62 - 74.

[12] Benjie Chen, Kyle Jamieson, Hari Balakrishnan and Robert Morris, "SPAN: an energy-efficient coordination algorithm for Topology maintenance in ad hoc wireless networks," *ACM Wireless Networks Journal*, Vol. 8, No. 5, Sep. 2002.

[13] Chih-fan Hsin and Mingyan Liu, "Network coverage using low duty-cycled sensors: random and coordinated sleep algorithms," *Proceedings of the third international symposium on Information processing in sensor networks*, Berkeley, California, USA.

[14] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Transactions On Information Theory*. Vol. 46, Mar. 2000.

[15] C. F. Chiasserini and M. Garetto, "Modeling the performance of wireless sensor networks," *IEEE Infocom2004*, Hong Kong, China, Mar. 2004.

[16] C. F. Chiasserini, R. Gaeta, M. Garetto, M. Gribaudo, D. Manini and M. Sereno, "Fluid models for large-scale wireless sensor networks," *Performance Evaluation*, Vol. 64, 2007, pp: 715 - 736.

[17] S. Shakkottai, R. Srikant and N. B. Shroff, "Unreliable sensor grids: coverage, connectivity and diameter," *IEEE INFOCOM*, San Francisco, CA, Apr. 2003.

[18] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," *Mobile Computing*, Imielinski and Korth, Eds., Vol. 353, Kluwer Academic Publishers, 1996.

[19] L.W. Fullerton, "Full duplex ultrawide-band communication system and method," *U.S. Patent 5 687 169*, Nov. 11, 1997.

[20] Nathaneil J. August, "Medium access control in impulse-based ultra wideband Ad Hoc and sensor networks," *Ph.D Dissertation, Virginia Polytechnic Institute and State University*, 2005.

[21] Tomaso Erseghe and Nicola Laurenti, "Design and performance evaluation of a full-duplex operating receiver for time-hopping UWB," *Mobile Network Application*, 2006, Vol. 11, pp: 429-439.

[22] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communication Magazine*, Aug. 2002, pp: 102 - 144.

[23] L. Kleinrock, "Queueing systems volume I: theory", A Wiley- Interscience Publication, 1975.

[24] Y. Wang, X. Li and Q. Zhang, "Efficient Algorithms for p-Self-Protection Problem in Static Wireless Sensor Networks ," *IEEE Transactions on Parallel and Distributed Systems*: Accepted for future publication.

[25] F. Ordonez, B. Krishnamachari, "Optimal information extraction in energy-limited wireless sensor networks ," *IEEE Journal on Selected Areas in Communications*, Vol. 22, Issue 6, Aug. 2004, pp: 1121 - 1129.

[26] Kurtis Kredo II and Prasant Mohapatra, "Medium access control in wireless sensor networks," *Computer Networks*, Vol. 51, 2007, pp: 961-994.

A Estimate the Neighborhood of Sensor Node

We have used a continuous time Markov chain with parameter α and β to characterize the behavior of the next-hops (i.e. the neighborhood of the reference node, see Fig.5). For a generic node k, if all its next-hops are in sleep mode, its neighborhood will be OFF with probability π^k_{off} and

$$\pi^k_{off} = \prod_{l \in H^k} P^l_s. \tag{38}$$

In (38), P^l_s can be easily obtained from (18); H^k is the set of all the next-hops of node k. For example, if node k has three next-hop nodes: n_1, n_2, n_3, then $H^k = \{n_1, n_2, n_3\}$. And the size of set H^k is 3. If we define z^k is the size of the set H^k, in the above example we have $z^k = 3$.

Note that in our sensor node model, the sleep duration is exponentially distributed with transition rate s, so we have,

$$\beta^k = z^k s. \tag{39}$$

It is then straightforward to estimate the other unknown transition rate as

$$\alpha^k = \frac{\pi^k_{off}}{1 - \pi^k_{off}} \beta^k. \tag{40}$$

And now we can calculate α^k, β^k directly based on the set H^k.

B Sensor Network Model

The whole sensor network can be considered as an *open queueing network*, and each packet stream originated from a node sensing the surrounding environment corresponds to the *external* arrival process to each queue.

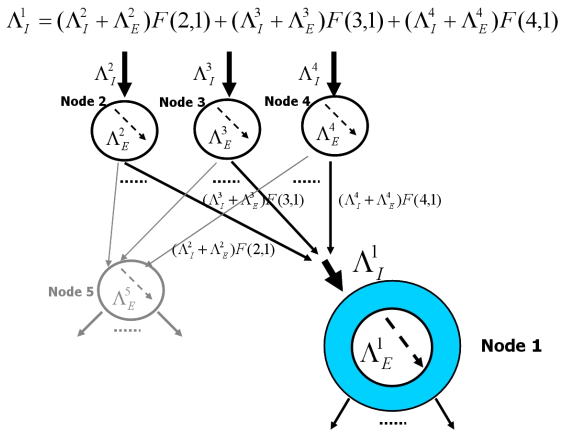

Figure 13: Network flow balance equation: example 1

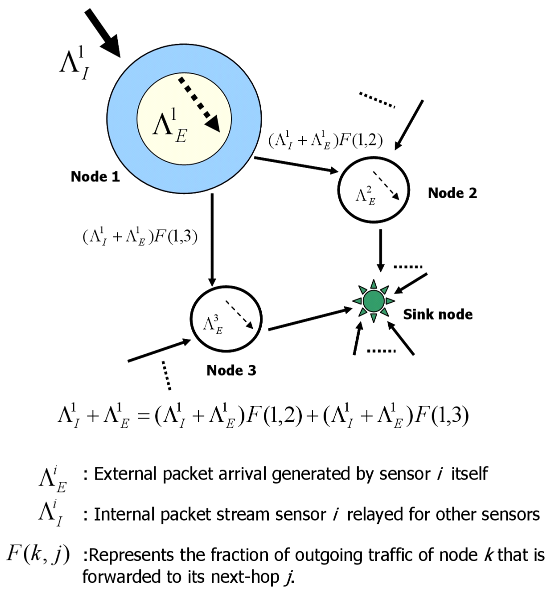

Figure 14: Network flow balance equation: example 2

We have defined Λ_E^k as the *equivalent packet generating rate* at node k from sensing the surrounding environment and

$$\Lambda_E^k = P_a \lambda_a^{e,k} + P_s \lambda_s^k. \tag{41}$$

In Section 3.1 we have formulated the sensor node behavior as a single server queue and mentioned that the *internal* packet arrival rate $\Lambda_I^k = \lambda_a^{i,k}$ at node k will be calculated through the following network model.

Now Λ_I^k can be obtained by solving the following network flow balance equation

$$\Lambda_I = (\Lambda_I + \Lambda_E)F, \tag{42}$$

where

$$\Lambda_I = [\Lambda_I^1, \Lambda_I^2, ..., \Lambda_I^N] = [\lambda_a^{i,1}, \lambda_a^{i,2}, ..., \lambda_a^{i,N}],$$

$$\Lambda_E = [\Lambda_E^1, \Lambda_E^2, ..., \Lambda_E^N],$$

respectively. And $\mathbf{F}$ is the matrix of transition probabilities between the queues in the open queueing network. Element $F(k, j)$ represents the fraction of outgoing traffic of node k that is forwarded to its next-hop j. We present two examples (Fig.13 and Fig.14) to illustrate the above flow balance equation.

In order to obtain $\mathbf{F}$, we should account for the routing policy adopted by the senor nodes as well as the joint active/sleep dynamics of all their next-hop nodes. As described in Section 3.2.1, the reference node always forwards its packets along the highest priority route associated with the lowest energy cost among all the available next-hops. Since we have assumed that the *active/sleep* dynamics of the next-hops are independent, we can build $\mathbf{F}$ by the following formula

$$F(k, j) = \theta(\prod_{n \in N_{k,j}} P_s^n) P_a^j, \tag{43}$$

where $N_{k,j}$ is the set of next-hop nodes that have higher priority than node j in the routing table of node k; and θ is a normalization factor such that the sum of $F(k, j)$ over all j's is equal to one. The formula of $F(k, j)$ means the probability that only node j is *Active* (which means node j can receive packets) and all the other next-hops with higher priority than node j are in *Sleep* mode.

C Derivation of Mean Packet Number

Define

$$M_{1,0}^a = \sum_{m=0}^{\infty} m p_{m,0}^a,$$

$$M_{1,1}^a = \sum_{m=0}^{\infty} m p_{m,1}^a,$$

$$M_{1,0}^s = \sum_{m=0}^{\infty} m p_{m,0}^s,$$

$$M_{1,1}^s = \sum_{m=0}^{\infty} m p_{m,1}^s.$$

From the above definitions, mean packet number in a sensor node is

$$\overline{M} = M_{1,0}^a + M_{1,1}^a + M_{1,0}^s + M_{1,1}^s. \tag{44}$$

In equation set (3), if we let each equation multiplies m, set $m = 0, 1, 2, ...\infty$, sum up them together, we can get the following equations about $M_{1,0}^a, M_{1,1}^a, M_{1,0}^s, M_{1,1}^s$,

$$\begin{cases} (\alpha + a)M_{1,0}^a - \beta M_{1,1}^a - s M_{1,0}^s = \lambda_a P_{a,0} - \mu P_{af} \\ (\beta + a)M_{1,1}^a - \alpha M_{1,0}^a - s M_{1,1}^s = \lambda_a P_{a,1} \\ -a M_{1,0}^a + (\alpha + s)M_{1,0}^s - \beta M_{1,1}^s = \lambda_s P_{s,0} \\ -a M_{1,1}^a + (\beta + s)M_{1,1}^s - \alpha M_{1,0}^s = \lambda_s P_{s,1}. \end{cases} \tag{45}$$

where

$$e_0 = [1, 0], e_1 = [0, 1],$$
$$P_{s,0} = e_0 \mathbf{P}^S,$$
$$P_{s,1} = e_1 \mathbf{P}^S.$$

Summing up the four equations in (45), we have

$$\lambda_a P_a + \lambda_s P_s = \mu P_{af}. \quad (46)$$

(46) means *in steady state, the equivalent packet arrival rate is equal to the equivalent packet service rate.*

The four equations in (45) are not independent, we need another equation to resolve the $M_{1,0}^a, M_{1,1}^a, M_{1,0}^s, M_{1,1}^s$. So define

$$M_{2,0}^a = \sum_{m=0}^{\infty} m^2 p_{m,0}^a,$$

$$M_{2,1}^a = \sum_{m=0}^{\infty} m^2 p_{m,1}^a,$$

$$M_{2,0}^s = \sum_{m=0}^{\infty} m^2 p_{m,0}^s,$$

$$M_{2,1}^s = \sum_{m=0}^{\infty} m^2 p_{m,1}^s.$$

From equation set (3) again, if we let each equation multiplies m^2, set $m = 0, 1, 2, ...\infty$, sum up them together, we can get the following equations about $M_{2,0}^a, M_{2,1}^a, M_{2,0}^s, M_{2,1}^s$,

$$\begin{cases} (\alpha + a)M_{2,0}^a - \beta M_{2,1}^a - sM_{2,0}^s - 2(\lambda_a - \mu)M_{1,0}^a = \\ \qquad \lambda_a P_{a,0} + \mu P_{af} \\ (\beta + a)M_{2,1}^a - \alpha M_{2,0}^a - sM_{2,1}^s - 2\lambda_a M_{1,1}^a = \lambda_a P_{a,1} \\ -aM_{2,0}^a + (\alpha + s)M_{2,0}^s - \beta M_{2,1}^s - 2\lambda_s M_{1,0}^s = \lambda_s P_{s,0} \\ -aM_{2,1}^a + (\beta + s)M_{2,1}^s - \alpha M_{2,0}^s - 2\lambda_s M_{1,1}^s = \lambda_s P_{s,1}. \end{cases} \quad (47)$$

Summing up the four equations in (47) and combing the result with (46) can result in an additional equation

$$(\lambda_a - \mu)M_{1,0}^a + \lambda_a M_{1,1}^a + \lambda_s M_{1,0}^s + \lambda_s M_{1,1}^s = -\mu P_{af}. \quad (48)$$

Combing (48) with the first three equations in (45), we can get

$$\begin{cases} (\alpha + a)M_{1,0}^a - \beta M_{1,1}^a - sM_{1,0}^s = \lambda_a P_{a,0} - \mu P_{af} \\ (\beta + a)M_{1,1}^a - \alpha M_{1,0}^a - sM_{1,1}^s = \lambda_a P_{a,1} \\ -aM_{1,0}^a + (\alpha + s)M_{1,0}^s - \beta M_{1,1}^s = \lambda_s P_{s,0} \\ (\lambda_a - \mu)M_{1,0}^a + \lambda_a M_{1,1}^a + \lambda_s M_{1,0}^s + \lambda_s M_{1,1}^s = -\mu P_{af}. \end{cases} \quad (49)$$

Define

$$M^1 = \begin{bmatrix} M_{1,0}^a \\ M_{1,1}^a \\ M_{1,0}^s \\ M_{1,1}^s \end{bmatrix},$$

equation set (49) can be written in matrix form as

$$\Omega M^1 = \Gamma \quad (50)$$

where

$$\Omega = \begin{bmatrix} (\alpha + a) & -\beta & -s & 0 \\ -\alpha & (\beta + a) & 0 & -s \\ -a & 0 & (\alpha + s) & -\beta \\ (\lambda_a - \mu) & \lambda_a & \lambda_s & \lambda_s \end{bmatrix}, \quad \Gamma = \begin{bmatrix} (\lambda_a P_{a,0} - \mu P_{af}) \\ \lambda_a P_{a,1} \\ \lambda_s P_{s,0} \\ -\mu P_{af} \end{bmatrix}.$$

Define $e^M = [1, 1, 1, 1]$, then the mean packet number is

$$\overline{M} = e^M \Omega^{-1} \Gamma. \quad (51)$$

D Analysis of Mean Delay

Define

$W_m(t) = Prob\{$waiting time $> t | m$ packets in queue; environment is ON;server is Acitve$\}$
$V_m(t) = Prob\{$waiting time $> t | m$ packets in queue; environment is OFF;server is Acitve$\}$
$O_m(t) = Prob\{$waiting time $> t | m$ packets in queue; environment is ON;server is Sleep$\}$
$H_m(t) = Prob\{$waiting time $> t | m$ packets in queue; environment is OFF;server is Sleep$\}$.

We have the following equation set

$$\begin{cases} \frac{dW_m(t)}{dt} = -(\mu + \alpha + a)W_m(t) + \mu W_{m-1}(t) + \alpha V_m(t) + aO_m(t) \\ \frac{dV_m(t)}{dt} = -(\beta + a)V_m(t) + \beta W_m(t) + aH_m(t) \\ \frac{dO_m(t)}{dt} = -(\alpha + s)O_m(t) + \alpha H_m(t) + sW_m(t) \\ \frac{dH_m(t)}{dt} = -(\beta + s)H_m(t) + \beta O_m(t) + sV_m(t). \end{cases} \quad (52)$$

In the above four equations, moving $'dt'$ to the right side and integrating the equations with respect to t from 0 to ∞ and using the fact that

$$W_m(\infty) = V_m(\infty) = O_m(\infty) = H_m(\infty) = 0$$
$$W_m(0) = V_m(0) = O_m(0) = H_m(0) = 1$$

to get

$$\begin{cases} (\mu + \alpha + a)\overline{W}_m = 1 + \mu \overline{W}_{m-1} + \alpha \overline{V}_m + a\overline{O}_m \\ (\beta + a)\overline{V}_m = 1 + \beta \overline{W}_m + a\overline{H}_m \\ (\alpha + s)\overline{O}_m = 1 + \alpha \overline{H}_m + s\overline{W}_m \\ (\beta + s)\overline{H}_m = 1 + \beta \overline{O}_m + s\overline{V}_m. \end{cases} \quad (53)$$

From (53) we can derive the recursive equations of $\overline{W}_m, \overline{V}_m, \overline{O}_m, \overline{H}_m$.