Hindawi Publishing Corporation
Mobile Information Systems
Volume 2016, Article ID 3085408, 9 pages
http://dx.doi.org/10.1155/2016/3085408

Research Article

Sleep Control Game for Wireless Sensor Networks

Sang Hoon Lee,[1] Hyeokman Kim,[2] and Lynn Choi[1]

[1] *School of Electrical Engineering, Korea University, Seoul 136-701, Republic of Korea*
[2] *School of Computer Science, Kookmin University, Seoul 136-702, Republic of Korea*

Correspondence should be addressed to Lynn Choi; lchoi@korea.ac.kr

Received 8 December 2015; Accepted 8 February 2016

Academic Editor: Lingjie Duan

In wireless sensor networks (WSNs), each node controls its sleep to reduce energy consumption without sacrificing message latency. In this paper we apply the game theory, which is a powerful tool that explains how each individual acts for his or her own economic benefit, to analyze the optimal sleep schedule for sensor nodes. We redefine this sleep control game as a modified version of the Prisoner's Dilemma. In the sleep control game, each node decides whether or not it wakes up for the cycle. Payoff functions of the sleep control game consider the expected traffic volume, network conditions, and the expected packet delay. According to the payoff function, each node selects the best wake-up strategy that may minimize the energy consumption and maintain the latency performance. To investigate the performance of our algorithm, we apply the sleep control game to X-MAC, which is one of the recent WSN MAC protocols. Our detailed packet level simulations confirm that the proposed algorithm can effectively reduce the energy consumption by removing unnecessary wake-up operations without loss of the latency performance.

1. Introduction

To reduce the energy consumption, each sensor node employs duty cycling where each node periodically sleeps [1]. Researches in earlier stage of duty cycling [1–4] assume that all the nodes in a network have the same static duty cycle whether or not nodes wake up at the same time. Since many applications of WSN assume random events [1–10], traffic varies each time. Therefore, it is hard for the static duty cycling to optimize wake-up schedule according to traffic condition. Dynamic duty cycle [5–10] allows each node to adjust its duty cycle on-demand. Therefore, nodes can reduce energy consumption for unnecessary wake-up operations if there is no traffic. However, the existing dynamic duty cycling schemes exploit a heuristic sleep control which may degrade energy efficiency and latency performance.

In this paper, we mathematically model and analyze the duty cycling of nodes to improve the energy efficiency of a MAC protocol without sacrificing the message latency. The communication activity of nodes is similar to human economic activities since both activities aim to maximize benefits—network performance for nodes versus economic profit for human—without global information for the entire network or market. In this regard, we adopt the game theory, which is a powerful tool that explains how each individual acts for his or her own economic benefit, as a tool for modeling and analyzing the duty cycle operation of sensor nodes. Since the communication activity of each node prioritizes over communication activities of other nodes and each node operates independently, the communication activity of a node can be regarded as the noncooperative game [11]. We redefine a sleep control game as the modified version of the Prisoner's Dilemma [11], which is one of the representative noncooperative games.

In the sleep control game, each node decides whether or not it wakes up for the cycle. Therefore, each node selects its strategy based on the payoff function. And, the wake-up probability is the key for each node's strategy in the sleep control game. Since the sleep state of a node affects both the energy consumption and the message latency, the wake-up probability of each node needs to consider the expected traffic volume, network conditions, and the expected packet delay, which are three components for the payoff function in the sleep control game. According to the payoff function, each node selects the best wake-up strategy which may minimize the energy consumption and maintain the latency performance. We show that the sleep control game is stable and it has Nash Equilibrium with the designed payoff function.

To investigate the performance of our algorithm, we apply the sleep control game to X-MAC [4] which is one of the recent WSN MAC protocols that employs the static duty cycling. From our detailed packet level simulations [12], we find that nodes with our algorithm can stay on sleep state 2.19 times longer than nodes with X-MAC. By reducing unnecessary idle listening, our algorithm saves 62% of the average per-node energy consumption of X-MAC. Nevertheless, our algorithm shows comparable latency performance to X-MAC. This suggests that our algorithm based on the mathematical model can provide an optimal dynamic duty cycling without relying on a heuristic. And, we believe that this is the first analytic framework for dynamic duty cycling.

2. Backgrounds and Related Works

A game theory describes and analyzes decision making process. In this paper, we limit our discussion to noncooperative model: the interaction between rational decision makers. The term rational decision makers here refer to those who are selfish and act for their best interest. The model described above is referred to as a "game," and the decision makers are called "players." This situation could be seen as follows: players choose a strategy from predefined list of strategies that will maximize their profit. A utility function would be deployed by each player to analyze another player's strategy selection. A normal form of a game G is given by $G = [N, A, \{u_i\}]$.

$N = \{1, 2, 3, \ldots\}$ is the set of players (decision makers), A_i is the strategy set of player i, $A = A_1 \times A_2 \times \cdots \times A_n$ is the Cartesian product of the set of strategies available to each player, and $\{u_i\} = \{u_1, \ldots, u_n\}$ is the set of utility functions that describes a measure of player's benefit that each player i wishes to maximize. The utility function u_i of each player i depends on the previous strategy selection, $\mathbf{a}_i$, and the other player's strategy selections, $\mathbf{a}_{-i}$. $\mathbf{a}_i$ and $\mathbf{a}_{-i}$ together make up a unique action tuple $\mathbf{a}$, which represents the action of each player. Mathematically $\mathbf{a}$ is the best response by player i to $\mathbf{a}_{-i}$ if $\mathbf{a} \in \{\arg\max u_i(\mathbf{a}_i, \mathbf{a}_{-i})\}$.

From the outcome of this model, we can conclude that stable states exist in the model. We figure these states are the Nash Equilibria. A system would be in Nash Equilibria when any individual player cannot increase profit by choosing any other strategies. In other words, Nash Equilibria could be expressed as the consistent projection of the outcomes of which there would be no incentive for each individual player to choose different strategies to maximize its profit.

However, although Nash Equilibria do exist in the model, Nash Equilibria do not mean the best outcome of a game. In many cases, Pareto optimality [11] is the representation of the efficiency of a product. An outcome could be expressed as Pareto optimal only when no other outcome can make every player well off while the outcome makes one player better off at least. Pareto optimality is formally defined as follows.

For all strategies $\mathbf{a}$, there is no strategy $\mathbf{b}$ such that $u_k(\mathbf{a}) > u_k(\mathbf{b})$ for some $k < N$ where N is the number of players.

There have been a few studies that apply game theory to MAC protocol design [13–16]. They are interested in increasing the network throughput by improving the contention resolution algorithm in the traditional wireless communication environment such as ad hoc network [13] and WLAN [14–16]. Thus, they focus on the improvement of the communication performance rather than the energy efficiency.

In WSNs, quite a few studies applied the game theory to the routing protocols [17, 18], power management techniques [19–24], and a backoff technique [25]. However, there is no game theoretic study for duty cycling of MAC protocols. Our sleep control game is the first study that applies the game theory to the duty cycle operations of MAC protocols. In the rest of this section we will review the state-of-the-art duty cycling techniques for WSNs.

Researches in earlier stage of duty cycling [1–4] assume that all the nodes in a network have the same static duty cycle whether or not nodes wake up at the same time. Since many applications of WSN assume random events [1–10], traffic varies each time. Therefore, it is hard for the static duty cycling to optimize wake-up schedule according to traffic condition. Dynamic duty cycle [5–10] allows each node to adjust its duty cycle on-demand. Therefore, nodes can reduce energy consumption for unnecessary wake-up operations if there is no traffic.

To our knowledge, AMAC [10] is the first sensor network MAC protocol that can support variable duty cycle operations. The main ideas underlying AMAC are twofold. First, each node can adjust the duration of the periodic interval depending on the network traffic. Second, a node can also adjust the duration of its active period depending on the traffic. This dynamic adjustment of both the active period and the periodic interval enables the duty cycle of each sensor node to adapt to the network traffic, resulting in significant energy savings for idle nodes and improved communication performance for busy nodes at the same time. However, the cycle time of a node needs to be adjusted according to a 2^n multiple of the minimum cycle time T. Since the latency requirement of a given application is not always a 2^n multiple of T, AMAC may overwork to guarantee the latency requirement.

A few more recent schemes [7–9] have studied the use of dynamic duty cycle operations. In PL-MAC [7] each node can reduce its duty cycle if its remaining energy is not enough to guarantee the predefined lifetime. If the expected lifetime of a node is longer than the predefined lifetime, a node may increase its duty cycle to reduce the communication latency. D-RMAC [8] focuses on the traffic condition. If a packet starts being buffered as traffic increases, a source node may double its duty cycle by transmitting an extra control packet to notify new duty cycle information. While AMAC, PL-MAC, and D-RMAC assume synchronous scheduling, MaxMAC [9] applies dynamic duty cycling to asynchronous scheduling, where each node wakes up independently. Like AMAC, each node can adjust its duty cycle depending on traffic.

While previous works [1–10] have focused on sensors' communication, ACDA [26] claims that the energy consumption of each sensor such as a camera should be considered. Authors modeled and simplified the scheduling problem under the consideration. ACDA controls both the lengths of a cycle time and sensing time to maximize the network utilization under periodic sensing application scenarios while our algorithm assumes nonperiodic event detection scenarios.

To exploit a heuristic sleep control, the schemes discussed so far require a predefined trigger condition such as traffic occurrence [8, 10], a packet rate [9], or the remaining energy [7]. Although these trigger conditions can be derived from a reverse engineering, it is very hard to prove that the trigger conditions should lead to an optimal wake-up schedule of sensor nodes.

3. Sleep Control Algorithm

3.1. Sleep Control Game: Game Theoretic Model of Sleep Control. Consider that each node has a set of N neighbor nodes in a contention based MAC protocol. We assume that each node independently controls its wake-up schedule and collects the network condition information within a single hop. The wireless channel is assumed to be error-free and packet loss is only due to collisions.

We model the interaction among wireless sensor nodes as a noncooperative game since each node operates independently and prioritizes its communication over neighbor's communications. Since each node is selfish and blind to the neighbor's strategies, the interaction model is similar to the Prisoner's Dilemma. We call the model as sleep control game.

We define the sleep control game by using the expected message latency, node's wake-up probability, contention measure, and the expected traffic volume. In practice, it is hard for a wireless sensor node to learn directly the wake-up probabilities of neighbors. Each node infers the neighbor's wake-up probabilities by observing the network condition of the previous cycle. In addition, it is hard to predict an end-to-end packet delay from a source to a destination. However, each node can estimate the packet delay by itself. The expected message latency can be estimated by using the wake-up probability, contention measure, and the expected traffic volume.

We assume that each node dynamically adjusts wake-up probability, w_i, in response to the expected latency performance, l_i. In other words, a next wake-up probability can be expressed as a function of a current wake-up probability and the expected latency performance. Hence, sleep control is a distributed, iterative feedback system mathematically given by

$$w_i(t+1) = F_i(w_i(t), l_i(t)), \quad l_i(t) = M_i(\mathbf{w}(t)), \tag{1}$$

where $w_i(t)$ is the wake-up probability of node i at time t, $\mathbf{w}(t) = [w_i(t)]$ is the corresponding vector, and $l_i(t)$ is the expected packet latency on node i. Note that a packet latency on a node depends on the sleep states of a node and its neighbor nodes. Therefore, $l_i(t)$ is a function of the sleep states that can be expressed as the vector $\mathbf{w}(t)$. Here, F_i models the sleep control algorithms and M_i models the latency update mechanisms. The expected packet latency is also affected by the traffic volume, which can be estimated from the buffer state and the event rate.

We assume that (1) has equilibrium (w, l). The fixed point of (1) defines an implicit relation between the equilibrium wake-up probability w_i and the expected packet latency l_i:

$$w_i = F_i(w_i, l_i). \tag{2}$$

If F_i is continuously differentiable and $\partial F_i/\partial l_i \neq 0$ in $[0, 1]$, then, by the implicit function theorem [27], there exists a unique continuously differentiable function f_i such that

$$l_i(t) = f_i(w(t)). \tag{3}$$

We define the utility function of each node i as

$$U_i(w_i) = \int f_i(w_i)\, dw_i. \tag{4}$$

Since $f_i(w_i) = l_i \geq 0$ and $U_i(\cdot)$ is an integral, $U_i(\cdot)$ is a monotonic function that is continuous and nondecreasing. It is reasonable to assume that $f_i(\cdot)$ is a decreasing function—the larger the wake-up probability, the smaller the expected packet latency. This implies that $U_i(\cdot)$ is strictly concave. With the above utility function, we define a sleep control game as follows.

Definition 1. A sleep control game G is defined as a quadruple $G = \{N, (S_i)_{i\in N}, (u_i)_{i\in N}, (l_i)_{i\in N}\}$, where N is a set of players (sensor nodes). For a player $i \in N$, its own wake-up probability is its strategy $S_i = \{w_i \mid w_i \in [\alpha_i, \beta_i]\}$ with $0 \leq \alpha_i \leq \beta_i \leq 1$. Payoff function $u_i(w)$ can be expressed by using utility function $U_i(w_i)$ and the expected packet latency $l_i = M_i(\mathbf{w})$:

$$u_i(w) = U_i(w_i) - \varepsilon w_i l_i. \tag{5}$$

The latency affection factor, ε, indicates the impact of the single hop latency on the end-to-end message latency, which is a variable constant according to the network condition.

The payoff function can be interpreted as the gain of utility from the packet latency discounted by the wake-up cost. One property of this game is that the computation of the payoff function does not require the explicit exchange of wake-up probability of each node among the nodes. Thus, this game can be played and implemented in a distributed manner. In addition, this game reduces the energy consumption and the bandwidth usage for transmitting control messages.

Since the strategy $w_i \in S_i$ is the wake-up probability of a node i, it is less than or equal to 1. The packet latency on node i is inversely proportional to w_i if there is no collision. Thus, the payoff function $u_i(\cdot)$ has a nice economic interpretation: the gain of utility from the packet latency is discounted by the wake-up cost. The sleep control game regards (1) as the strategy update algorithm to find equilibrium. Therefore, we can specify the equilibrium properties of a sleep control algorithm by using the utility function $U_i(w_i)$ and the latency performance l_i. In other words, by exploiting these factors, we can define the sleep control game whose equilibrium determines the steady state properties such as the latency performance and the energy performance. (F, M) can specify the adaptation of the wake-up probability and suggest different strategies to approach the equilibrium of the game.

3.2. Analyzing Transmission Latency. In WSNs, the packet latency is influenced by node's buffer state, traffic volume, and the wake-up state of a receiver node. When a node i transmits a packet p to node j, the packet latency l_i on node i is given by

$$l_i = P_{ij}(p) + Q_i(p), \tag{6}$$

where $P_{ij}(p)$ denotes the delay due to the sleep state of nodes i and j and $Q_i(p)$ denotes the delay due to transmission failure. If we assume that the probability for an event occurrence is δ, the probability that the packet p collides with another packet is given by

$$\begin{gathered} C(p) = \left(1 - (1-\delta)^{Ad-1}\right), \\ A = 4\int_0^r \int_0^{r+x/2} \sqrt{r^2 - (r-y)^2}\, dy\, dx, \end{gathered} \tag{7}$$

where d denotes the node density, r denotes the transmission range of a node, and A denotes the average area of union of transmission range of adjacent two nodes. If we assume that each node transmits a single packet during a single cycle, packet p needs to be delayed for a single cycle when there is a collision. Therefore, the delay due to a collision, $Q_i(p)$, can be given by

$$Q_i(p) = T + \sum_{k\in N, k\neq i} kT(C(p))^k, \tag{8}$$

where T denotes the minimum cycle time which is longer than the time to carry out a single packet transmission, which consists of the time for transmitting a data packet and control packets and the control time space such as Interframe Spacing.

The sleep delay, $P_{ij}(p)$, is influenced by two wake-up probabilities: sender's wake-up probability $w_i(t)$ and receiver's wake-up probability $w_j(t)$. Since two nodes must wake up for a successful communication, the probability for the occurrence of the sleep delay is $(1 - w_i(t)w_j(t))$. Therefore, $P_{ij}(p)$ is given by

$$P_{ij}(p) = T\left(m + \left(1 - w_i(t)\, w_j(t)\right)\right), \tag{9}$$

where m denotes the delayed cycle time due to the sleep state after the packet p arrives at node i.

3.3. Designing Payoff Function. Equation (1) defines the probability for a node to wake up by using the previous wake-up probability and the expected packet latency. If we do not consider the previous wake-up probability, the wake-up probability of each node may be biased toward 0 or 1. This will lead to long message latency due to sleep delay when the probability stands near 0. Or, it may lead to high energy consumption due to idle listening when the probability stands near 1. Therefore, we expend the wake-up probability for a node as

$$w_i(t+1) = \begin{cases} 1, & (P_i(t) = 1), \\ F_i(w_i(t), l_i(t)), & (P_i(t) = 0). \end{cases} \tag{10}$$

$P(t)$ denotes whether or not the node transmits a packet in the previous cycle. If $P(t)$ is 1, the node transmitted a packet. By considering the previous communication, we can effectively reduce the sleep delay when there is burst traffic or a message consists of multiple packets.

By using (5) and (6), we can derive the payoff function as

$$\begin{aligned} u_i(w) &= \int f_i(w_i)\, dw_i - \varepsilon w_i l_i = \int l_i(t)\, dw_i - \varepsilon w_i l_i \\ &= Q_i(p)\, w_i + T\left((m+1)\, w_i - \frac{w_i^2 w_j}{2}\right) - \varepsilon w_i l_i. \end{aligned} \tag{11}$$

According to (11), each node requires receiver's wake-up probability w_j. However, it may be difficult for a node to directly acquire the wake-up probability of the receiver. Therefore, each node infers the neighbor's wake-up probabilities by observing the network condition of the previous cycle. According to (10), if a node i overheard the node j's communication in the previous cycle, i assumes that j should wake up in this cycle. On the other hand, if i slept or did not overhear j's communication, i assumes that $w_i \approx w_j$ by assuming nodes are evenly deployed and event occurs randomly anywhere.

3.4. Equilibrium of Sleep Control Game. The wake-up probability needs to be stable and unique in order to find the best strategy. To verify the stability of wake-up probability, we should analyze the equilibrium of the sleep control game and show that the equilibrium is unique. By proving the following three theorems, we will show the existence (Theorem 3) and the uniqueness (Theorems 5 and 7) of the equilibrium. We use the Nash Equilibrium [11] to find the equilibrium.

We denote the strategy (wake-up probability) selection for node i by w_i. We further denote the strategy selection for all nodes but node i by $\mathbf{w}_{-i} = (w_1, \ldots, w_{i-1}, w_{i+1}, \ldots, w_N)$ and use $(w_i, \mathbf{w}_{-i})$ for the strategy profile $(w_1, \ldots, w_{i-1}, w_i, w_{i+1}, \ldots, w_N)$ for all the nodes in a network. A vector of wake-up probability $\mathbf{w}^*$ is a Nash Equilibrium if, for all the nodes $i \in N$, $u_i(w_i^*, \mathbf{w}_{-i}^*) \geq u_i(w_i, \mathbf{w}_{-i}^*)$ for all $w_i \in S_i$. We see that the Nash Equilibrium is a set of strategies for which no player has an incentive to change unilaterally.

Assumption 2. The utility function $U_i(\cdot)$ is continuously differentiable, strictly concave, and with finite curvatures that are bounded away from zero; that is, there exist some constants μ and λ, such that $1/\mu \geq -1/U_i''(w_i) \geq 1/\lambda > 0$.

Theorem 3. *Under Assumption 2, there exists a Nash Equilibrium for sleep control game G.*

Proof. Assumption 2 is a standard assumption in economics. As shown in (4), $U_i(\cdot)$ is strictly concave since we assume that $f_i(\cdot)$ is a decreasing function. In other words, for any

x and y in the interval and for any t in $[0,1]$, $U_i(\cdot)$ follows the strictly concave condition: $U_i(tx+(1-t)y) > tU_i(x) + (1-t)U_i(y)$. The strategy space $[0,1]$ is a bounded closed set. In one-dimensional Euclidean space, a bounded closed set is a compact convex set. Since the strategy spaces S_i are compact convex sets, and the payoff functions are continuous and concave in w_i, there exists a Nash Equilibrium [11].

Since payoff function $u_i(\cdot)$ is concave in w_i, at the Nash Equilibrium, w_i^* satisfies

$$\left(U_i'\left(w_i^*\right) - l_i\left(\mathbf{w}^*\right)\right)\left(w_i - w_i^*\right) \le 0, \quad \forall w_i \in S_i. \tag{12}$$

Define a function $V(\mathbf{w}) = \sum_{i\in N}(U_i(w_i) - w_i) - \prod_{i\in N}(1 - w_i)$. Then, this equation becomes an optimality condition for the following optimization problem [28]:

$$\max_{w\in S_1\times S_2\times\cdots\times S_N} V(\mathbf{w}). \tag{13}$$

That is, the Nash Equilibria of the wake-up game are optimal points of problem (13). □

Assumption 4. Let $\gamma(\mathbf{w}) = \prod_{i\in N}(1 - w_i)$ and denote the smallest eigenvalue of $\nabla^2\gamma(\mathbf{w})$ over $\mathbf{w}$ by $v_{\min}$. Then, $-\mu - v_{\min} < 0$.

Theorem 5. *Under Assumptions 2 and 4, the sleep control game G has a unique Nash Equilibrium.*

Proof. The Hessian of function $V(\mathbf{w})$ is written as

$$\begin{aligned}\nabla^2 V(\mathbf{w}) &= \operatorname{diag}\left(\left[U_1''(w_1), \ldots, U_N''(w_N)\right]\right) \\ &\quad - \nabla^2\gamma(\mathbf{w}),\end{aligned} \tag{14}$$

where $\operatorname{diag}([U_1''(w_1), \ldots, U_N''(w_N)]) \le -\mu\mathbf{I}$ and $\nabla^2\gamma(\mathbf{w}) \ge v_{\min}\mathbf{I}$. Thus, under Assumption 4,

$$\nabla^2 V(\mathbf{w}) \le \left(-\mu - v_{\min}\right)\mathbf{I} < 0. \tag{15}$$

By second-order conditions [28], $V(\mathbf{w})$ is a strictly concave function over the strategy space. So, the optimization problem (13) has a unique optimal, and the sleep control game has a unique Nash Equilibrium.

The equilibrium condition (12) implies that, at the Nash Equilibrium, w_i^* either takes value at the boundaries of the strategy space $[0,1]$—that usually leads to the longest packet latency (selecting "0") or to the most energy consumption (selecting "1")—or satisfies

$$U_i'\left(w_i^*\right) = l_i\left(\mathbf{w}^*\right). \tag{16}$$

We call a Nash Equilibrium a nontrivial equilibrium $\mathbf{w}^*$ if, for all nodes i, $\mathbf{w}^*$ satisfies (16). Otherwise, it is a trivial equilibrium. According to (16), at nontrivial equilibrium,

$$\Gamma_i\left(w_i^*\right) = \gamma\left(\mathbf{w}^*\right). \tag{17}$$

Note that (17) is independent of i. Thus, $\Gamma_i(w_i^*) = \Gamma_j(w_j^*)$ for any $i, j \in N$. □

Assumption 6. $\Gamma_i(w_i^*) = (1 - w_i)(1 - U_i'(w_i))$, $i \in N$, are all strictly increasing or all strictly decreasing.

Theorem 7. *If the control game G has a nontrivial Nash Equilibrium, it must be unique.*

Proof. Suppose that there are two nontrivial Nash Equilibriums $\mathbf{w}_1$ and $\mathbf{w}_2$. From (17) we require that there exist $\gamma_1, \gamma_2 > 0$ such that, for all i,

$$\begin{aligned}\Gamma_i\left(w_{1i}\right) &= \gamma_1, \\ \Gamma_i\left(w_{2i}\right) &= \gamma_2.\end{aligned} \tag{18}$$

Since $\Gamma_i(w_i)$ is one-to-one, $\gamma_1 \ne \gamma_2$. Without loss of generality, assume that $\Gamma_i(w_i)$ is increasing and $\gamma_1 > \gamma_2$. Thus, $\mathbf{w}_1 > \mathbf{w}_2$ for all i. By (16), $U_i'(\mathbf{w}_{1i}) = l_i(\mathbf{w}_1) > l_i(\mathbf{w}_2) = U_i{}'(\mathbf{w}_{2i})$, which contradicts the fact that $U_i'(w_i)$ is a decreasing function. Thus, the sleep control game G has a unique nontrivial Nash Equilibrium. □

Each node i can choose any utility function $U_i(\cdot)$ as appropriate. If all the nodes have the same utility function, the system is said to have homogeneous users. If the nodes have different utility functions, the system is said to have heterogeneous users. The motivation for studying systems of heterogeneous users is to provide differentiated services to different wireless nodes. To this end, we further differentiate between symmetric and asymmetric equilibria as follows.

A Nash Equilibrium $\mathbf{w}^*$ is said to be a symmetric equilibrium if $\mathbf{w}_i^* = \mathbf{w}_j^*$ for all i and j. Otherwise, it is an asymmetric equilibrium. Since by symmetry there must be multiple asymmetric Nash Equilibria if there exists any, the following result follows directly from Theorems 5 and 7.

For a system of homogeneous users, suppose Assumption 4 or Assumption 6 holds. If the sleep control game has a nontrivial Nash Equilibrium, it must be unique and symmetric. More generally, for a system with several classes of homogeneous users, under the same assumption, if the sleep control game has a nontrivial Nash Equilibrium, it must be unique and symmetric. This corollary guarantees the uniqueness of nontrivial Nash Equilibrium, and it guarantees maximum energy-delay product of nodes. This will facilitate the design of medium access control. Since at trivial Nash Equilibrium some player takes a strategy (wake-up probability) at the boundary of the strategy space, a trivial Nash Equilibrium usually has great unfairness or low payoff. So, nontrivial Nash Equilibrium is desired. If there is no nontrivial Nash Equilibrium, we may need to look for alternative solution other than the Nash Equilibrium. For example, we may use Nash bargaining framework in cooperative game theory [11] to derive a desired equilibrium solution.

4. Simulation and Results

In this section we analyze the performance of our algorithm by applying the sleep control game to X-MAC [4] which is one of the representative WSN MAC protocols recently proposed. X-MAC exploits the asynchronous wake-up scheduling that nodes independently wake up with a static duty cycle.

Table 1: Simulation parameters.

Parameters	
Topology of a network	400 random nodes
Simulation time	1 hour
Data packet size	100 bytes
Number of packets for an event	10 packets
Basic cycle time of X-MAC, $4T$	300 ms (5% duty cycle)
Event rate	0.1% per T
Power consumption (Tx, Rx, idle)	30 mW, 15 mW, 15 mW
Latency affection factor, ε	1
Maximum distance from a sink node	10 hops

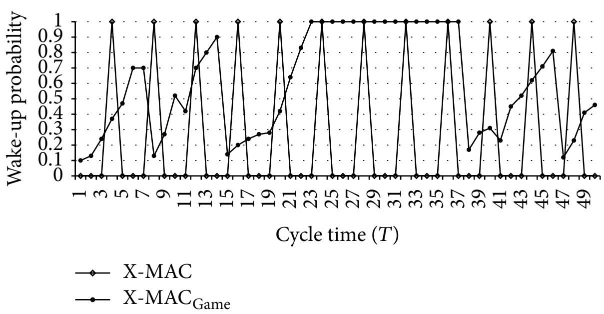

Figure 1: The wake-up probability according to time elapse.

We implemented X-MAC and a modified version of X-MAC with our sleep control algorithm, on NS-2 [12]. With the detailed packet level simulations, we evaluate the performance of our algorithm from various perspectives: the energy consumption, the message delay, the network throughput, and the wake-up probability. The simulation parameters are shown in Table 1. We assume that a sink node locates at the center of the network field where 400 nodes locate randomly. We also assume that an event occurs at random position and the nearest node detects the event. In other words, we evaluate the performance of our algorithm under a nonperiodic event scenario model that can be applied to environmental monitoring applications such as fire surveillance, mechanical malfunctions, and biochemical hazard.

4.1. Wake-Up Probability. Figure 1 shows the wake-up probability of a sensor node by varying the node cycle time. Since the basic cycle time of X-MAC is $4T$, a node with X-MAC has $3T$ sleep period. In X-MAC, each node wakes up according to the predefined schedule. The wake-up probability of a node for each T is 0 or 1. Unlike X-MAC, X-MAC$_{\text{Game}}$ uses the probabilistic wake-up approach. Starting from a sleep state, the wake-up probability of a node is gradually increased until the node actually wakes up.

According to the results, a node with X-MAC$_{\text{Game}}$ wakes up only 2 times during $20T$ while a node with X-MAC wakes up 5 times. The average sleep period of X-MAC$_{\text{Game}}$ is $6.58T$ while X-MAC has $3T$ sleep period. In other words, X-MAC$_{\text{Game}}$ can reduce the number of unnecessary wake-up operations when there is no traffic.

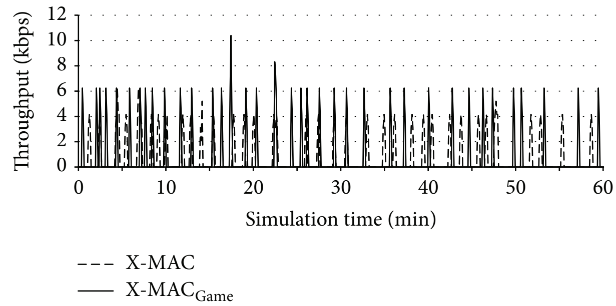

Figure 2: The network throughput according to time elapse.

Note that a node with X-MAC$_{\text{Game}}$ maintains the wake-up state when the cycle time changes from $23T$ to $37T$. These continuous wake-up states represent that the node participates in communications. The reason why the X-MAC$_{\text{Game}}$ node consumes $15T$ cycles to deliver a single 10-packet message is that the node suffered from transmission failures and the sleep delay due to the sleep state of a receiver.

4.2. Latency Performance. Figure 2 shows the network throughput as time passes. Since nodes which participate in communications wake up at every minimum cycle time T in X-MAC$_{\text{Game}}$, X-MAC$_{\text{Game}}$ can deliver more packets than X-MAC during the same interval. It means that X-MAC$_{\text{Game}}$ can effectively deliver burst traffic while X-MAC suffers from the accumulation of the sleep delay. According to the results, both X-MAC and X-MAC$_{\text{Game}}$ show a uniform distribution of the network throughput since we generate events randomly over the simulation time. When there are multiple events, the throughputs of two protocols reached a peak, which is proportional to the number of simultaneous transmissions.

Figure 3 shows the average message latency according to the hop distance from a source to a destination. As shown in Figure 1, the average sleep period of X-MAC$_{\text{Game}}$ is $6.58T$ while X-MAC has $3T$ sleep period. This suggests that the sleep control game can substantially reduce the idle listening overhead as much as 54.5% by eliminating unnecessary wake-up operations. However, the average message latency of X-MAC$_{\text{Game}}$ is slightly longer than that of X-MAC. Since nodes in X-MAC$_{\text{Game}}$ can adjust their wake-up schedules according to the traffic condition, they can adaptively decide their wake-up either to reduce idle listening or to reduce the sleep delay depending on traffic.

Figure 4 shows the distribution of the packet latency of single hop communications. Since X-MAC uses asynchronous wake-up scheduling with 300 ms cycle time, the difference between the wake-up times of two nodes causes the packet latency to be concentrated at the interval between 0.3 s and 0.5 s. Except the first packet of a message, other packets may suffer from the buffering delay. Therefore, the buffering delay makes X-MAC to have the second major concentration interval of [0.6 s, 0.85 s].

Like X-MAC, X-MAC$_{\text{Game}}$ has two concentration intervals: [0.1 s, 0.2 s] and [0.4 s, 0.7 s]. Since X-MAC$_{\text{Game}}$ nodes increase the sleep period when there is no traffic, the sleep

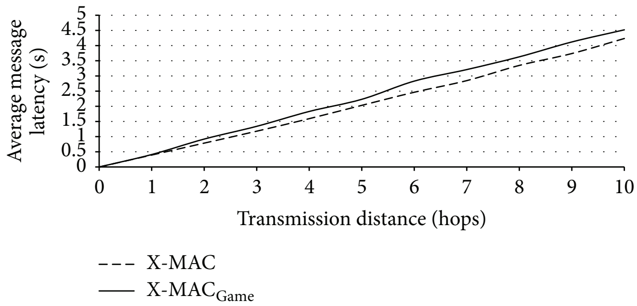

Figure 3: The average message latency according to hop counts.

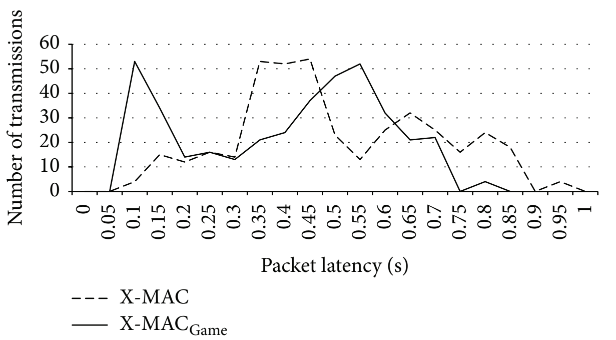

Figure 4: The distribution of the packet latency for a single hop communication.

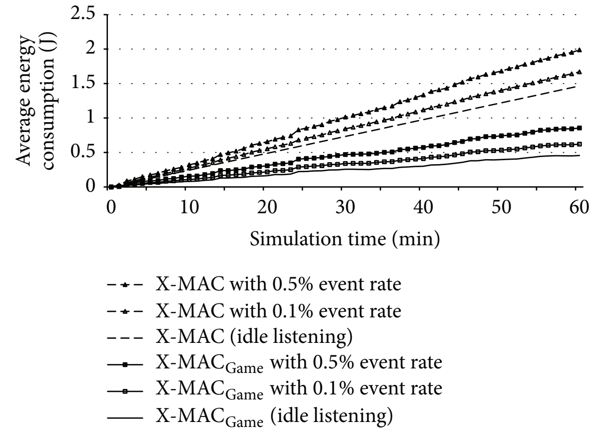

Figure 5: The average energy consumption according to time elapse.

delay before transmitting the first packet of a message increases. Therefore, the latencies for the first packet of a message concentrate on [0.4 s, 0.7 s]. In X-MAC$_{\text{Game}}$, each node that participates in a communication wakes up at every minimum cycle time T until the communication ends. This continuous wake-up state results in shorter buffering delay than X-MAC. Therefore, the most of the packet latencies are distributed between 0.4 s and 0.7 s.

When a source generates multiple messages, the latencies of the following messages can be reduced since nodes on the common routing path already wake up after reporting the first message. Therefore, X-MAC$_{\text{Game}}$ has the second concentration period of [0.1 s, 0.2 s]. This second concentration period may be dominant as the number of simultaneous events increases.

According to the results of Figures 3 and 4, X-MAC$_{\text{Game}}$ has comparable performance while it keeps lower duty cycle than X-MAC. Moreover, X-MAC$_{\text{Game}}$ can effectively deal with burst traffic, which is common traffic pattern in WSNs. This suggests that the sleep control game can effectively reduce energy consumption due to unnecessary wake-up operations without sacrificing the communication performance.

4.3. Energy Consumption. Figure 5 shows the average per-node energy consumption as time passes. From our simulations we find that the communication traffic is not the dominant source of the energy consumption. Even with heavy traffic, idle listening is still the dominant energy waste source [1]. Our results show that X-MAC consumes 87.7% of the total dissipated energy for idle listening when the event rate is 0.1% per T. The energy consumption due to idle listening accounts for 73.7% of the total when we increase the event rate by 0.5% per T.

When there is no traffic, X-MAC$_{\text{Game}}$ consumes only 31.1% energy of X-MAC. Since this result shows the accumulation of the average energy consumption, the slope of graph varies according to the duty cycle of a sensor node. In addition, the dynamic wake-up scheduling approach of X-MAC$_{\text{Game}}$ also reduces the energy consumption due to overhearing since the approach reduces the chance for adjacent nodes to wake up at the same cycle by waking up only the nodes in a communication path more frequently. When the event rate is 0.1% per T, X-MAC$_{\text{Game}}$ consumes additional 0.165 J per node to deliver the traffic while X-MAC requires 0.205 J per node. X-MAC$_{\text{Game}}$ can save 19.5% of the energy consumption for communications by removing energy wastes due to overhearing.

4.4. Impact of the Latency Affection Factor. Figure 6 shows the average message latency as we vary the latency affection factor, ε. According to (18), the wake-up probability is inversely proportional to the latency affection factor. In other words, we can reduce the message latency by reducing the latency affection factor. However, if we reduce the factor, the wake-up probability increases, and nodes will consume more energy for idle listening.

Figure 7 shows the average energy consumption according to the latency affection factor. As we discussed above, the small latency affection factor increases the wake-up probability. Therefore, the number of wake-up operations and the average energy consumption per node increase.

As shown in Figures 6 and 7, there is a tradeoff between message latency and energy consumption. Figure 8 shows the energy-delay product according to the latency affection factor, ε. Note that the latency affection factor is influenced by various environmental conditions such as the number of neighbors, event rate, the performance requirements of an application, and channel quality. In our experiments, the optimal latency affection factor is 2.37 from the performance viewpoint. However, the optimal latency affection factor is

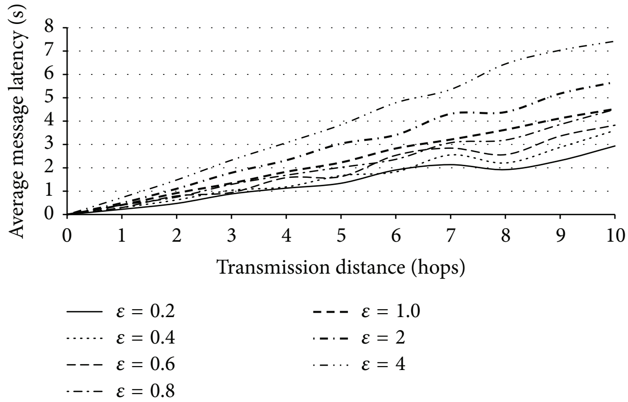

Figure 6: The average message latency according to the latency affection factor, ε.

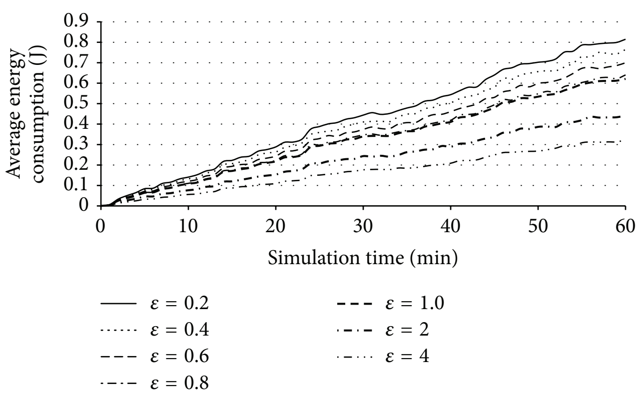

Figure 7: The average energy consumption according to the latency affection factor, ε.

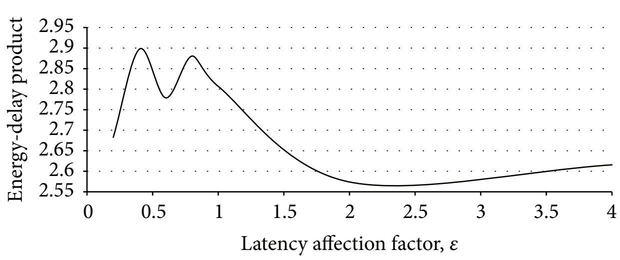

Figure 8: Energy-delay product according to the latency affection factor.

not always the best selection. If we assume that the message latency requirement of an application is bigger than 10 s in these simulations, we can save more energy by setting ε as 4 while we do not violate the message latency requirement. Therefore, the reverse engineering approach would be suitable for adjusting the latency affection factor.

5. Conclusions

In this paper, we introduce a novel game theoretic MAC approach to improve the energy efficiency of WSNs. Our scheme adjusts nodes' wake-up schedules dynamically by exploiting the wake-up probability based on the game theory, which is an efficient tool for analyzing the interaction between multiple independent rational players. To model the interaction between sensor nodes, we redefine a sleep control game as a modified version of the Prisoner's Dilemma. Payoff function of the sleep control game considers the expected traffic volume, network condition, and the expected message latency.

The major contribution of this paper is that it introduces a mathematical approach to control the duty cycle of a sensor node while the conventional researches exploit a heuristic approach. With the detailed packet level simulations, we confirm that the sleep control game can effectively reduce energy consumption without sacrificing the network performance. Furthermore, the sleep control game can effectively deal with burst traffic and it is also suitable for massive sensor network with inexpensive sensor nodes.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean Government (MSIP) (no. NRF-2015R1A2A1A16074932). This research was also supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2015-R0992-15-1012) supervised by the IITP (Institute for Information & Communications Technology Promotion).

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, 2004.

[2] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proceedings of the Second International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 95–107, November 2004.

[3] C. C. Enz, A. El-Hoiydi, J.-D. Decotignie, and V. Peiris, "WiseNET: an ultralow-power wireless sensor network solution," *Computer*, vol. 37, no. 8, pp. 62–70, 2004.

[4] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys '06)*, pp. 307–320, Boulder, Colo, USA, November 2006.

[5] T. Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*, pp. 171–180, ACM, Los Angeles, Calif, USA, November 2003.

[6] P. Lin, C. Qiao, and X. Wang, "Medium access control with a dynamic duty cycle for sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '04)*, pp. 1534–1539, Atlanta, Ga, USA, March 2004.

[7] A. Irandoost, S. Taheri, and A. Movaghar, "PL-MAC: prolonging network lifetime with a MAC layer approach in wireless sensor networks," in *Proceedings of the 2nd International Conference on Sensor Technologies and Applications (SENSORCOMM '08)*, pp. 109–114, Cap Estérel, France, August 2008.

[8] Y. I. Kim, B.-G. Choi, C. S. Kim, and M. Y. Chung, "A synchronized MAC protocol considering packet generation intervals in wireless sensor networks," in *Proceedings of the IEEE Region 10 Conference (TENCON '09)*, pp. 1–5, Singapore, November 2009.

[9] P. Hurni and T. Braun, "MaxMAC: a maximally traffic-adaptive MAC protocol for wireless sensor networks," in *Wireless Sensor Networks*, vol. 5970 of *Lecture Notes in Computer Science*, pp. 289–305, Springer, Berlin, Germany, 2010.

[10] S. H. Lee, J. H. Park, and L. Choi, "AMAC: traffic-adaptive sensor network MAC protocol through variable duty-cycle operations," in *Proceedings of the IEEE International Conference on Communications (ICC '07)*, pp. 3259–3264, IEEE, Glasgow, UK, June 2007.

[11] D. Fudenberg and J. Tirole, *Game Theory*, The MIT Press, 1991.

[12] The Network Simulator—ns-2, http://www.isi.edu/nsnam/ns/.

[13] V. S. Borkar and A. A. Kherani, "Random access in wireless ad hoc networks as a distributed game," in *Proceedings of the Workshop on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOPT '04)*, Cambridge, UK, March 2004.

[14] M. Čagalj, S. Ganeriwal, I. Aad, and J.-P. Hubaux, "On selfish behavior in CSMA/CA networks," in *Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '05)*, pp. 2513–2524, IEEE, Miami, Fla, USA, March 2005.

[15] L. Chen, T. Cui, S. H. Low, and J. C. Doyle, "A game-theoretic model for medium access control," in *Proceedings of the International Wireless Internet Conference (WICON '07)*, Austin, Tex, USA, October 2007.

[16] L. Chen, S. H. Low, and J. C. Doyle, "Random access game and medium access control design," *IEEE/ACM Transactions on Networking*, vol. 18, no. 4, pp. 1303–1316, 2010.

[17] D. Miller, S. Tilak, and T. Fountain, "Token equilibria in sensor networks with multiple sponsors," in *Proceedings of the 1st Workshop on Stochasticity in Distributed Systems (StoDiS '05)*, San Jose, Calif, USA, December 2005.

[18] R. Kannan and S. S. Iyengar, "Game-theoretic models for reliable path-length and energy-constrained routing with data aggregation in wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 6, pp. 1141–1150, 2004.

[19] A. Agah, S. K. Das, and K. A. Basu, "A game theory based approach for security in wireless sensor networks," in *Proceedings of the IEEE International Conference on Performance, Computing, and Communications (IPCCC '04)*, pp. 259–263, IEEE, Phoenix, Ariz, USA, April 2004.

[20] S. Sengupta and M. Chatterjee, "Distributed power control in sensor networks: a game theoretic approach," in *Distributed Computing—IWDC 2004: 6th International Workshop, Kolkata, India, December 27–30, 2004. Proceedings*, vol. 3326 of *Lecture Notes in Computer Science*, pp. 508–519, Springer, Berlin, Germany, 2004.

[21] X. Zhang, Y. Cai, and H. Zhang, "A game-theoretic dynamic power management policy on wireless sensor network," in *Proceedings of the International Conference on Communication Technology (ICCT '06)*, pp. 1–4, IEEE, Guilin, China, November 2006.

[22] E. Campos-Nañez, A. Garcia, and C. Li, "A game-theoretic approach to efficient power management in sensor networks," *Operations Research*, vol. 56, no. 3, pp. 552–561, 2008.

[23] D. Niyato, E. Hossain, and A. Fallahi, "Sleep and wakeup strategies in solar-powered wireless sensor/mesh networks: performance analysis and optimization," *IEEE Transactions on Mobile Computing*, vol. 6, no. 2, pp. 221–236, 2007.

[24] S. K. Jayaweera and K. Hakim, "A cooperative game theoretic solution for lifetime maximization of WSNs in sequential estimation," in *Proceedings of the 5th International Conference on Information and Automation for Sustainability (ICIAfS '10)*, pp. 233–238, Colombo, Sri Lanka, December 2010.

[25] K. S. Kwak and S. Mehta, "An energy-efficient MAC protocol in wireless sensor networks: a game theoretic approach," *Eurasip Journal on Wireless Communications and Networking*, vol. 2010, Article ID 926420, 2010.

[26] J. He, L. Duan, F. Hou, P. Cheng, and J. Chen, "Multiperiod scheduling for wireless sensor networks: a distributed consensus approach," *IEEE Transactions on Signal Processing*, vol. 63, no. 7, pp. 1651–1663, 2015.

[27] R. Abraham, J. E. Marsden, and T. Ratiu, *Manifolds, Manifolds, Tensors, Analysis, and Applications*, vol. 75 of *Applied Mathematical Sciences*, Springer, New York, NY, USA, 2nd edition, 1988.

[28] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation*, Prentice Hall, New York, NY, USA, 1989.