Cooperative Scheduling for Adaptive Duty Cycling in Asynchronous Sensor Networks

Jianhui Zhang[1], Feng Xia[2], Shaojie Tang[3], Xingfa Shen[1] and Bei Zhao[1,4]

[1] College of Computer Science and Technology, Hangzhou Dianzi University, 310018, China.
[2] School of Software, Dalian University of Technology, 116620, China
[3] College of Science and Technology, Temple University, Philadelphia, PA19122 USA
[4] College of Biomedical Engineering and Instrument Science, Zhejiang University, Hangzhou, 310058, China
Email: jhzhang@ieee.org (Corresponding Author)

To support sustainable operation of Wireless Sensor Networks (WSNs) with limited energy, duty cycling is promising to be a popular solution. However, it is a big challenge to guarantee that neighboring nodes have common active time for communication, and fairly and efficiently share their common channels under duty cycling, especially when networks are asynchronous or duty cycle is extremely low. Existing LPL- and contention-based protocols are not energy-efficient and cannot ensure high channel utility. Additionally, synchronization-based MAC protocols suffer from extra energy cost and low synchronization precision. This paper presents a Localized and On-Demand scheme (LOD) for duty cycle adjustment based on a specifically designed Semi-Quorum System (SQS). LOD can adaptively adjust duty cycle of each node according to its load so as to avoid channel contention, thus achieving high channel utilization and fairness for channel access within asynchronous networks. For simplicity, a sensor network with a tree infrastructure is constructed to evaluate LOD. The extensive experiments is also conducted on a real test-bed of 100 TelosB nodes to evaluate the performance of LOD. Comparing with B-MAC, LOD substantially reduces contention for the channel access and accordingly the energy consumption, and significantly improves the network throughput.

Keywords: Duty Cycle; Semi-Quorum System; Medium Access Control; Wireless Sensor Networks

Received 00 October 2013; revised 00 00 2013

1. INTRODUCTION

Energy and channel resource are two of main constraints in Wireless Sensor Networks (WSNs). Duty cycling is one of main ways to save energy and increase channel utilization in WSNs [1]. In duty cycled WSNs, a necessary task is to cooperate the communication among nodes efficiently. When each node has its own active mode and duty cycle different from others, it brings challenges to finish this task. WSNs are essentially asynchronous because of those unprecise clocks equipped on sensor nodes, which makes the challenges bigger. Although there are some existing ways, such as TDMA-based protocols, to cooperate the communication within networks [2][3], they need synchronization, which suffers from extra energy cost and low precision, and is not effective in real applications, such as GreenOrbs [4], because of the temperature variation in natural environment [3]. CSMA-based protocol can avoid energy consumption on synchronization, but it costs much channel resource and energy on contention for channel access, which results in low channel utilization. The case is even worse when duty cycle is extremely low.

Furthermore, sensor nodes need adjust their own duty cycles in real applications because they have different communication loads. A proper duty cycle of a node can finish its communication load and must be as low as possible. It is a challenge problem how to adaptively adjust the duty cycle of a node according to its load especially when networks are asynchronous and communication cooperation among nodes is necessary.

Therefore, a challenging problem is to increase channel utility without synchronization under duty cycling. This paper designs a Localized and On-Demand scheme (LOD) for duty cycle adjustment based on a specifically designed Semi-Quorum System (SQS), denoted by $\hat{\mathcal{Q}}$. LOD can adaptively adjust duty cycle of each node according to its load and schedule the active time of every node while no synchronization and channel contention are adopted within asynchronous networks. Thus it can achieve higher channel utilization and fairness for channel access than previous MAC protocols under duty cycling. Energy efficiency under LOD is correspondingly higher than those under other protocols. The key properties of LOD owe to SQS, which is designed in this paper based on Quorum System (QS) [5] and inherits the advantages of some QSs on the non-empty intersection and rotation closure properties. QS is applied to establish channel control in dynamic spectrum access networks [6], to save power [7], to maximize throughput in limited information multiparty MAC with QS [8] and to schedule duty-cycling [9]. However, SQS is more flexible and can ensure each node to be active in less time than QS.

In LOD, we introduce a concept *demand* $\mathcal{D}$ to represent the amount of data that a node has to transmit or receive in each period T, which composes of m time slots. Each node u will be active in a set of time slots $\varsigma \subset T$ according to its demand so it can select a quorum, which contains ς. In the rest time slots T/ς, u will sleep to save energy. Thus the duty cycle of u is $|\varsigma|/|T|$, which means that the duty cycle of u is adaptively adjusted according to its demand. *The intersection property* of SQS ensures that any pair of neighboring nodes have common active time to communicate with each other. *The rotation closure property* of SQS guarantees that any pair of neighboring nodes can have rendezvous active time to communicate with each other without adopting any synchronization protocol within asynchronous networks. Our paper gives the contributions as follows:

- We design a new quorum system, SQS, satisfying nonempty intersection property and rotation closure properties. SQS has high flexible and can obtain lower duty cycle than QS with same load.
- Based on SQS, a new protocol LOD is designed to adaptively adjust the duty cycle of each node according to its demand. Thus the fairness of channel access and channel utilization are increased comparing to the existing contention-based MAC protocols.
- LOD guarantees each pair of neighboring nodes have proper common active time to communicate with each other without synchronization. Therefore, energy and time are saved comparing with contention- and synchronization-based MAC protocols.
- A quorum selection method is proposed to allocate quorums for each node thus the worst case of channel utilization is bounded under LOD and confliction of active time is avoided within networks.

We also evaluate the performance of LOD on a real test-bed consisting of 100 TelosB nodes.

The organization of this paper is as follows. Section 2 presents the network model, formulate our problem and introduce the QS and SQS technologies. In Section 3, we design the new system SQS and analyze its properties. Section 5 presents the designing of our scheme LOD and evaluation of its preliminary properties. Meanwhile, the performances of LOD are presented when the certain demand is implemented in asynchronous networks. In Section 6, we implement our protocol on our real test-bed composed by 100 TelosB nodes and show the performance of LOD with the experimental results. Section 7 tells the related works in recent years in duty cycling and MAC designing. The work of whole paper is concluded in Section 8.

2. SYSTEM MODEL AND PRELIMINARY

2.1. Network Model

A network is composed of a set V of n nodes and a set E of edges among these nodes. We formulate the network by a graph $G(V, E)$. Each node has a unique ID and s denotes the sink. The radius of a graph G with respect to s, denote by R, is the maximum distance (hop) between s and the nodes in G. Each node has same receiving power P_r and listening P_l, and different transmission power P_t with others. Each node has its transmission range $\mathbb{r}$ and interference range ρ. This paper studies the effect of the duty cycle adjustment on the delay under several popular interference models (denoted by $\mathcal{M}$): RTS/CTS ($\mathcal{M}_{rc}$), protocol model ($\mathcal{M}_{pr}$) and physical model ($\mathcal{M}_{ph}$) [10]. Under $\mathcal{M}_{pr}$, a node v_i can successfully transmit to a neighbor v_j *iff* the condition $||v_i - v_j|| \leq \mathbb{r}_i$ is satisfied and other transmitter v_k simultaneously transmitting with v_i should be sufficiently apart from v_j, *i.e.* $||v_k - v_j|| \geq (1+\delta')||v_i - v_j||$, where $\delta' > 0$ is a constant. Under $\mathcal{M}_{rc}$, if every pair of transmitter v_i and receiver v_j can successfully communicate with each other *iff* $||v_i - v_j|| \leq \mathbb{r}_i$ and all nodes within the interference region of v_i and v_j cannot transmit simultaneously. Under $\mathcal{M}_{ph}$, a node v_i can successfully transmit to a neighbor v_j when $SINR = \frac{P_i \cdot d_{ij}^{-\alpha}}{B \cdot N_0 + \sum_{k \in I} P_k d_{kj}^{-\alpha}} \geq \eta$, and η is the SINR threshold, d_{ij} (or d_{kj}) is the distance between the nodes v_i (or v_k) and v_j, N_0 is the background noise, P_i and P_k are the transmission power, I is a index set of nodes simultaneously transmitting and α is the path lost exponent.

Based on $G(V, E)$, a tree, denoted by $\mathscr{T}$, can be constructed. The tree rooted at the sink s, from which the tree is ranked into $\mathscr{L}$ levels. The level which the sink locates at is labeled ι_0. The parent and the children of a node u is denoted by $p(u)$ and $c(u)$ respectively.

2.2. Problem Statement

In WSNs with tree infrastructure, the communication is restricted between parent nodes and their children nodes when there is no contention for media access. In many applications, such as canopy closure estimation [4], WSNs work under the well designed or independent duty cycle mode as discussed in Section 1 in order to save energy. Notice that a network cannot achieve high channel utilization under fixed duty cycle because the duty cycle of a node is determined by its location in a network and the task the network undertakes.

Suppose that a period T is composed of m time slots, *i.e.* $T = \{\tau_1, \cdots, \tau_m\}$. Each node u is set to be active in a subset of time slots $\varsigma_u \subseteq T$. A pair of neighboring nodes can communicate with each other only when they are active in some common time slots. Additionally, the clock of each node u is not precise and results in a clock shift δ_u, which is often unknown to u itself in advance. Thus the subset of time slots ς_u accordingly has a shift in each period, *i.e.* $\varsigma'_u = \varsigma_u + \{\delta_u\}$. Suppose a node v is the parent of two nodes x and y. Therefore, when nodes must accomplish their demands in asynchronous networks, the follows equations should be satisfied.

$$\begin{aligned}\forall x, y \in c(u): &\varsigma'_v \cap \varsigma'_x \geq t(\mathcal{D}_{v,x}), \varsigma'_v \cap \varsigma'_y \geq t(\mathcal{D}_{v,y})\\ &\varsigma'_x \cap \varsigma'_y = \varnothing, \varsigma'_v, \varsigma'_x, \varsigma'_y \subset T \qquad (1)\end{aligned}$$

where $t(\mathcal{D}_{v,x}) = \max\{t(\mathcal{D}_v), t(\mathcal{D}_x)\}$ and $t(\mathcal{D}_v)$ and $t(\mathcal{D}_x)$ are respectively the time spent on accomplishing $\mathcal{D}_v$ and $\mathcal{D}_x$ between v and x. Equation (1) means that each pair of neighboring nodes must have common active time enough to accomplish demands.

On the other hand, nodes must adjust their duty cycle as low as possible to save energy. Therefore, the problem we study is as follows: in each period T, each node u locally chooses a subset $\varsigma'_u \subset T$ to guarantee any arbitrary pair of neighboring nodes satisfying Equation (1) while the energy consumption is saved. This paper aims to analyzing the *demand condition* under which the demand of each node can be accomplished when it selects a time slot set according to its demand. Under this condition, we design a new scheme LOD to assign a set of time slots for each node so that Equation (1) can be satisfied. The specific demand, data aggregation, is implemented to evaluate the performance of LOD on the load, which is the frequency that each time slot is used.

2.3. Quorum Systems

We first introduce a basic tool QS [11], denoted by $\mathcal{Q}$.

Definition 2.1. *Given a period T (a set of time slots) to be a universal set. A QS $\mathcal{Q} \subset 2^T$ is a set of subsets of T such that every two subsets intersect, i.e. $Q_1 \cap Q_2 \neq \varnothing$ where Q_1, $Q_2 \in \mathcal{Q}$ are called* quorums.

Denote the quorum allocated to a node u by Q_u. $Q_u \subset \mathcal{Q}$ contains a subset of time slots $\varsigma_u \subset T$ according to Definition 2.1. The cardinality of ς_u is denoted by κ_u, *i.e.* $\kappa_u = |\varsigma_u|$.

Definition 2.2. *A* rotation *of the quorum Q is defined by $\mathscr{S}(Q, i)$=$\{(\tau_j + i)$ mod $m|\tau_j \in Q\}$, where i is a non-negative integer and Q is a quorum in a QS $\mathcal{Q}$ under the universal T and $m = |T|$.*

Definition 2.3. *A QS $\mathcal{Q}$ satisfies the* rotation closure property *in T if any two quorums Q_1 and Q_2, $Q_1, Q_2 \in \mathcal{Q}$, satisfy the condition: $\forall i \in \{0, \cdots, m-1\}$: $Q_1 \cap \mathscr{S}(Q_2, i) \neq \varnothing$.*

In this paper, we design a new QS, called semi-QS (SQS), which is defined as follows.

Definition 2.4. *A SQS $\hat{\mathcal{Q}}$ is a set of subsets of the universal set T and is composed of a* basic subset *Q_b, and other non-basic subset. Thus every* normal subset *Q_1 intersects with the basic subset Q_b, i.e. $Q_b \cap Q_1 \neq \varnothing$ and does not intersects with any other non-basic subset, where $Q_1 \in \hat{\mathcal{Q}}$.*

After the definition of SQS is given, we can respectively call a basic and normal subset as Basic Semi-Quorum (BSQ), denoted by Q_b, and Normal Semi-Quorum (NSQ), denoted by Q_n.

3. SQS CONSTRUCTION

This section presents the construction of SQS based on B-QS [12]. Notice that SQS can also designed based on other QSs, such as grid, torus and cyclic QS only if they satisfy the rotation closure property [13]. Based on SQS, LOD is designed to adaptively adjust the active time of each node in two steps. In the first step, a tree is constructed $\mathscr{T}$. In the second step, acitve time slots are assigned to each node based on the tree. Finally, some properties of LOD are analyzed.

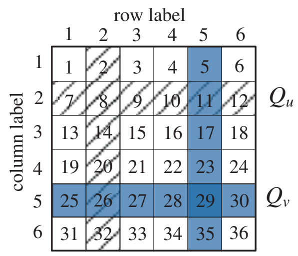


FIGURE 1. A grid QS $\mathcal{Q}_g$ contains T. There are $\lceil\sqrt{m}\rceil$ rows and columns in $\mathcal{Q}_g$.

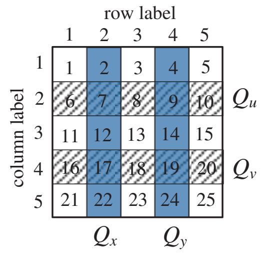


FIGURE 2. A SQS.

3.1. Designing SQS

Suppose a parent node v has two child nodes x and y. In WSNs with tree-type infrastructure (see the example in Figure 5), the child nodes x and y need not have rendezvous time when the networks implements some kind of tasks, such as data aggregation and collection. To satisfy this requirement, we design a SQS, which contains BSQs and NSQs, *i.e.* SQS={BSQs and NSQs}. In SQS, child nodes can select NSQs and parent node can select BSQs. There are several kinds of QS, such as Grid-QS and Torus-QS. Thus we can correspondingly design several kinds of SQS.

This paper only investigates the Grid QS (see the example in Figure 1) because it satisfies the rotation closure and non-empty intersection properties. Based on Grid QS, we can design Grid-SQS as shown in the example in Figure 2. A SQS contains m elements distributed in a rectangle with $R \times L$ grids. In a Grid-SQS, $R = L = \sqrt{m}$, and a BSQ is a set of a full row elements while a NSQ is a set of a full column elements.

In the example of Figure 2, the Grid SQS contains four semi-quorums: Q_u, Q_v, Q_x and Q_y, where Q_u Q_v are two BSQs and Q_x and Q_y are two NSQs. $Q_u = \{6, 7, 8, 9, 10\}$; $Q_v = \{16, 17, 18, 19, 20\}$; $Q_x = \{2, 7, 12, 17, 22\}$; $Q_y = \{4, 9, 14, 19, 24\}$. SQS has its own specific properties including non-empty intersection, rotation closure, adaptivity of duty cycle and quorum load.

Non-empty Intersection QSs, such as grid-QS, has non-empty intersection property. Inspired by the property, we introduce new property in order to assign active time among nodes when we design SQS. The new property keeps the empty intersection among NSQs or BSQs and the non-empty intersection between a NSQ and BSQ. According to the method of constructing SQS in the subsection 3.1, any two different NSQ can not occupy same columns. Thus we can obtain Lemma 3.1 as the example shown in Figure 2. Similarly, we can obtain Lemma 3.2.

Lemma 3.1. *The intersection between any pair of NSQs or BSQs is empty in SQS.*

Lemma 3.2. *There is at least one rendezvous element between a NSQ and BSQ in SQS.*

For example, there are no intersection between Q_y and Q_x and between Q_u and Q_v. Q_u and Q_x intersect on the element 7 in Figure 2;

Rotation Closure Another important property of QS is the rotation closure property, which is beneficial to design a QS-based protocol in asynchronous networks. According to Lemma 3.1, the rotation closure property of SQS is different from previous QSs on the property so a new rotation closure property in a Grid-SQS can be obtained.

Lemma 3.3. *In a Grid-SQS, any pair of NSQ and BSQ satisfy the rotation closure property.*

Proof. Let $\hat{\mathcal{Q}}$ be a Grid-SQS and Q_b be a BSQ and Q_n be a NSQ, where $Q_b, Q_n \in \hat{\mathcal{Q}}$. Without loss of generality, we suppose Q_b contains all elements on the row a of the array, namely $a + i$, where $1 \le a \le R$ and $i = 0, \cdots, \sqrt{m} - 1$. Suppose Q_n contains all elements on the column c of the array, namely $c + (j-1)\sqrt{m}$, where $1 \le c \le L$ and $j = 1, \cdots, \sqrt{m}$. Notice that $R = L = \sqrt{m}$ in Grid-SQS. Thus there is a common element $c + (a-1)\sqrt{m}$ in both Q_b and Q_n.

Then the rotation of Q_b and Q_n are respectively $\mathscr{S}(Q_b, i_r)$ and $\mathscr{S}(Q_n, j_r)$, where i_r and j_r are two non-negative integers. Notice that $\mathscr{S}(Q_b, i_r) = \{a + i + i_r \ mod \ m\}$ and $\mathscr{S}(Q_n, j_r) = \{c + (j-1)\sqrt{m} + j_r \ mod \ m\}$. Meanwhile, $\mathscr{S}(Q_n, j_r) = \{c + (j-1)\sqrt{m} + j_r \ mod \ m\} = \{(c + j_r) \ mod \ m + (j-1)\sqrt{m}\}$ must still contain a full column when j_r is given as an arbitrary non-integer. Although the elements contained in $\mathscr{S}(Q_b, i_r)$ fall into two rows when $i_r \ mod \ m \ne 0$, there are no two elements having same column label, *i.e.*, each column contains one element which belongs to $\mathscr{S}(Q_b, i_r)$. Therefore, $\mathscr{S}(Q_b, i_r)$ must have non-empty intersection with $\mathscr{S}(Q_n, j_r)$. □

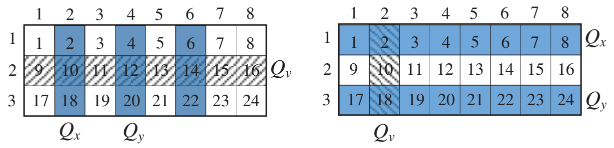
Duty Cycle In a tree network, a node can have adaptive duty cycle under SQS, which is different from previous QSs. A duty cycle of a node is the ratio of its active time to the whole period. In QS and SQS, each node sets a quorum to be active time in each period. Since a quorum contains at least one column and one row in grid-QS, the duty cycle is at least $\frac{2\sqrt{m}-1}{m}$. The duty cycle is at least $\sqrt{\frac{2}{m}}$ in cyclic-QS. The duty cycle can be adjustable in a SQS and be as small as $\frac{1}{m}$. In WSNs, an important requirement is that duty cycle of each node can be adjusted to minimize and balance the energy consumption among a network. Previous QSs cannot satisfy the requirement but SQS can by the following property.

3.2. Transmutation of SQS

In order to make SQS more adaptive, we discuss the transformation property of SQS since each node may have different amount of neighbors and tasks. The transformation property of SQS will obtain some new properties on increasing energy efficiency because parent nodes generally consume more power than their child nodes while it still satisfies the non-empty intersection and rotation closure properties. We first present the transformation of Grid SQS and call it as Rectangle Grid SQS (RG-SQS).

Grid SQS can be transformed into two kinds of RG-SQS (see the example in Figure 3). One is a row dominated transformation (see a example in Figure 3(a)) and the other is a column dominated transformation (see a example in Figure 3(b)) . In the row dominated RG-SQS, the cardinality of any BSQ is bigger than that of any NSQ. It is opposite in the column dominated RG-SQS.

In RG-SQS, the non-empty intersection and rotation closure properties are still satisfied between any pair of NSQ and BSQ.



(a) Row dominated (b) Column dominated

FIGURE 3. RG-SQS. Q_v is a BSQ.

4. LOD DESIGN

WSNs usually affords different kinds of tasks determined by application scenarios. So each node must afford different amount of demand correspondingly. The time to accomplish a node's demand is different from others' because of its location in a network and transmission rate. Thus duty-cycle should be adjustable online in practice. In this section, we design a new method LOD to allow each node locally decide its active time by cooperating with its neighboring nodes under asynchronization.

4.1. Designing LOD

LOD composes of two steps: tree construction and duty cycle designing.

Step I: For illustration, this paper constructs a minimum spanning tree (MST) $\mathscr{T}$ based on $G(V, E)$. In process of the tree construction, each node is assigned a Level label ι and collects the information of its one-hop neighbor nodes, which contains the IDs and Levels of other nodes in u's communication set in the tree. After the tree $\mathscr{T}$ is constructed, each node u should be assigned a period T_u to be active. Each period contains some consecutive integers and the cardinalities of all periods for all nodes are same. Here we assign each pair of neighboring nodes u and v with different periods T_u and T_v. Notice that only the non-leaf nodes can be assigned periods. We say two periods T_u and T_v are different if the difference between them is non-zero and positive integral multiple of $|T|$. Thus the period assignment problem is a *vertex coloring* problem. The chromatic number of $\mathscr{T}$, denoted by $\chi(\mathscr{T})$, is the smallest number of colors required by any vertex coloring of $\mathscr{T}$. It is Minimum Vertex Coloring problem to seek a vertex of a given network with the fewest colors. It is NP-hard in general [14].

$\chi(\mathscr{T})$ is determined by the interference model $\mathcal{M}$ under synchronization. It will be discussed under asynchronization in the next section. We need at least $\chi(\mathcal{M})$ to color all the regions and the regions with same color are conflict-free. We label each region σ_u with a color index θ_{σ_u}, $\theta_{\sigma_u} \in \mathscr{C} = \{1, \cdots, \chi(\mathcal{M})\}$.

Step II: This paper designs time schedule protocol to let each node adaptively set its active time slots enough to accomplish its demand while as low as possible to save energy. Notice that the network life time is necessarily not prolonged when the energy consumption of each single node is decreased. Thus we try not only to save the energy of each node but also to balance the energy consumption among parents and their child nodes in order to prolong network life and satisfy Equation (1). Here we suppose the network life is the period from point a network starting running to that the first node using up its energy in it.

A node's demand composes of transmission or receiving data. We introduce a parameter ζ for a parent node to indicate the ratio of its transmission data to receiving data. We denote the transmission data of a parent u by $\mathcal{D}_{uv}$, where v is a child node of u, and the receiving data of u by $\mathcal{D}_v$. Notice that the receiving data of u is the sum of data transmitted from its child node v. Thus $\zeta = \frac{\mathcal{D}_{uv}}{\mathcal{D}_v}$.

After each node knows its demand, it can adjust its duty cycle with RG-SQS by cooperating with its neighboring nodes. At **Step I**, each no-leaf node u is assigned a period T_u. Each node then adjusts its duty cycle base on RG-SQS by sharing its SQS $\hat{\mathcal{Q}}_u$ with its child nodes as shown in Figure 5. In this figure, v shares its SQS with its child nodes x and y. So a non-leaf node is often included in two SQSs as shown in Figure 5.

LEMMA 4.1. *The non-leaf nodes are covered by two SQSs and the leaf nodes are covered by the SQS of its parent.*

Lemma 4.1 indicates that each node need limited and little memory space to store SQSs. A SQS actually covers a group of nodes including a parent node and its child nodes. We can call a group of nodes covered by a SQS as a *basic group*. A network is composed by many such basic groups. In the follows of this paper, we use basic group as a model to analyze the active time scheduling.

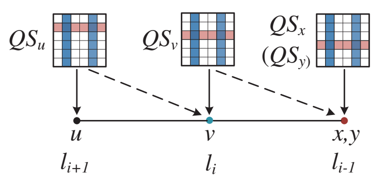


FIGURE 4. One node has multi-QS. One node covers at most two SQSs.

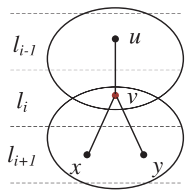


FIGURE 5. One communicate set overs two QSs.

In this step, the amount of each node's active time slots and scheduling for these slots are determined in each period. The active time of a node u is determined by not only its demand $\mathcal{D}_u$ but also the data rate ϱ_u and retransmission probability. In a real network, the data rate ϱ_u of each node u can be determined by previous methods [15]. Notice that ϱ_u also includes the effect of the packet loss rate. The retransmission actually makes a nodes to implement extra demand. This paper don't consider how to determine the extra demand, which was previously discussed [16], and use a parameter $\varphi \geq 1$ to indicate a node has to implement extra demand because of retransmission. Notice that we analyze the energy consumption under the basic group model in the follows. That means the energy consumption is consumed on the communication between a parent node and its child nodes. A child node u has demand $\mathcal{D}_u$ and a parent node v has demand $\mathcal{D}_{vu}$ corresponding to its child node u. Thus a child node u should adjust its active time to be at least $\frac{\varphi_u(\mathcal{D}_u+\mathcal{D}_{vu})}{\varrho_u}$. A parent node v should adjust its active time to be at least $\frac{\varphi_w}{\varrho}\sum\limits_{w\in c(v)\cup\{v\}}(\mathcal{D}_w+\mathcal{D}_{vw})$, where $\varrho=\min_{w\in c(v)\cup\{v\}}\varrho_w$.

This paper considers the demands of the child nodes, which have the same parent, are same with each other in a period, such as data aggregation or flooding and so on. Thus the demand of the parent node u is same corresponding to its different child nodes, *i.e.* $\mathcal{D}_{uv}=\mathcal{D}_{uw}$, where v and w are u's child nodes. In previous works, a node transmits a beacon message at the beginning [17] or both beginning and end [18] of an active slot in order to find its neighbors, and preambles in some MAC protocols [19] in order to establish communication with its neighbors. In this paper, the RG-SQS is designed to not only discover a node's neighbors and establish communication in much lower energy cost but also to set proper active time to finish the communication demand between each pair of nodes.

Each RG-SQS composes of R BSQs or L NSQs and $R\times L=m$. R and L are adjustable, for example, R and L are adjusted into different values in the three sub-figures of Figure 6. Under the definition of the network lifetime which is a period from the network starting running to the time when the first node uses up its energy, we should decrease the energy consumption of the node which consumes energy fastest in order to prolong the network life. Figure 6 gives an example to show that it can not prolong the network lifetime by decreasing the energy consumption of each node or a group of nodes. When a sensor node is in the active mode, its radio can transmit or receive data or listen. The RF power of a sensor node, such as TelosB and MICA2, is adjustable from -24dBm to 0dBm [20]. When a sensor node is in active mode, the energy consumption is not ignorable, for example the current draw of TelosB is 1.8mA under active mode. Thus we can describe the total energy consumption E_T of a parent node u and its child nodes $c(u)$ in each period by Equation (2). Notice that we discuss the energy consumption on communication.

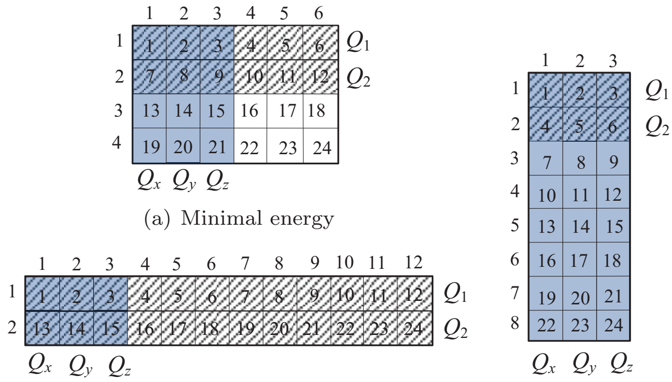
$$\begin{aligned}E_T &= \sum_{v\in c(u)\cup\{u\}} P_t^v t(\mathcal{D}_v) + P_r \sum_{v\in c(u)\cup\{u\}} t(\mathcal{D}_v)\\ &\quad + P_l \sum_{v\in c(u)\cup\{u\}} t_l^v \qquad (2)\end{aligned}$$

Where P_t^v is the transmission power of the node v. t_l^v is the total listening time of v in each period. $t(\mathcal{D}_v)$ denotes the time to finish $\mathcal{D}_v$.

TABLE 1. Energy consumption in different RG-SQS of Figure 6 and the unit is τV

Node	u	x	y	z	Total energy consumption
Figure6(b)	$436.8\tau V$	$32.8\tau V$	$32.8\tau V$	$32.8\tau V$	$535.2\tau V$
Figure6(c)	$98.4\tau V$	$145.6\tau V$	$145.6\tau V$	$145.6\tau V$	$535.2\tau V$
Figure6(a)	$211.2\tau V$	$70.4\tau V$	$70.4\tau V$	$70.4\tau V$	$422.4\tau V$

Notice to minimize the energy consumption, such as Equation (2), is not necessarily able to prolong the network life time. Figure 6 illustrates the energy consumption by the example of TelosB node, which uses Chipcon CC2420 radio. The current consumption of TelosB under receiving, listen and transmit mode with power -5dBm are 18.8mA, 18.8mA and 14mA. We suppose a period composes of 24 time slots *slot*, and a parent node u has three child nodes x, y and z. u selects two BSQs Q_1 and Q_2. x, y and z respectively select the NSQs Q_x, Q_y and Q_z. If the energy consumption of a single node, such as u or x, the RG-SQS should be transformed into the one in Figure 6(b) or Figure 6(c). In the common time slots between Q_1, Q_2 and Q_x, Q_y, Q_z, such as time slots 1, 2, 3, 13, 14 and 15, u communicates with x, y and z. In other time slots (uncommon time slots), the radio of each node listens or sleeps. Without loss of generality, we suppose the four nodes spend one half common time to receive and other half common time to transmit. Thus in the RG-SQSs of Figure 6,the energy consumption of different nodes is illustrated in Table 1.



(a) Minimal energy

(b) Row dominated

(c) Column dominated

FIGURE 6. To minimize the energy consumption does not necessarily lead to prolong the network life time.

As shown in Table 1, the total energy consumption in the RG-SQS of Figure 6(a) is minimal but the network lifetime is $\min\{\frac{1}{211.2},\frac{1}{70.4},\frac{1}{70.4},\frac{1}{70.4}\}=\frac{1}{211.2}$ if the total energy that each node initially has is $1\tau V$. In the RG-SQS of Figure6(b), the energy consumption of x, y and z is minimized but the network lifetime is $\min\{\frac{1}{436.8},\frac{1}{32.8},\frac{1}{32.8},\frac{1}{32.8}\}=\frac{1}{436.8}$. Thus the energy minimization maybe cannot be obtained when the network lifetime is maximized. In this paper, we try to minimize the maximal energy consumption of each node as described in Equation (3) so that the network lifetime is maximized.

$$\min \max_{v \in c(u) \cup \{u\}} E_v \tag{3}$$

$$\begin{aligned} s.t. \quad & R_u \times L_u = m \\ & L_u \geq d_u \\ & 1 \leq R_u \leq m, 1 \leq L_u \leq m \end{aligned} \tag{4}$$

Suppose parent node u selects r_u BSQs and a child node v of u selects l_v NSQs. Notice each child node should select at least one NSQ so Equation (4) should be satisfied. Equation (3) actually aims to decrease the energy consumption of each node by minimizing $r_u + l_v$. The energy consumption E_u of a parent node u is described in Equation (5).

$$\begin{aligned} E_u &= P_t^u \sum_{v \in c(u)} t(\mathcal{D}_{uv}) + P_r \sum_{v \in c(u)} t(\mathcal{D}_v) + P_l t_l^u \\ &= P_t^u d_u t(\mathcal{D}_{uv}) + P_r d_u t(\mathcal{D}_v) + P_l t_l^u \\ &= P_t^u d_u t(\zeta \mathcal{D}_v) + P_r d_u t(\mathcal{D}_v) + P_l t_l^u \\ &= P_t^u d_u \frac{\varphi_v}{\varrho} \zeta \mathcal{D}_v + P_r d_u \frac{\varphi_v}{\varrho} \mathcal{D}_v + P_l t_l^u \end{aligned} \tag{5}$$

The energy consumption E_v of a child node v is described in Equation (6).

$$\begin{aligned} E_v &= P_t^v t(\mathcal{D}_v) + P_r t(\mathcal{D}_{uv}) + P_l t_l^v \\ &= P_t^v t(\mathcal{D}_v) + P_r t(\zeta \mathcal{D}_v) + P_l t_l^v \\ &= P_t^v \frac{\varphi_v}{\varrho} \mathcal{D}_v + P_r \frac{\varphi_v}{\varrho} \zeta \mathcal{D}_v + P_l t_l^v \end{aligned} \tag{6}$$

Notice that we consider that child nodes, having a same parent node, have same demand in each period. Thus the active time of a child node v is $\frac{\varphi_v}{\varrho}\mathcal{D}_v + \frac{\varphi_v}{\varrho}\zeta\mathcal{D}_v = \frac{\varphi_v}{\varrho}\mathcal{D}_v(1+\zeta)$, which equals to that of other child node w, *i.e.* $\frac{\varphi_v}{\varrho}\mathcal{D}_v(1+\zeta) = \frac{\varphi_w}{\varrho}\mathcal{D}_w(1+\zeta)$. It means that the child nodes select same amount of columns in a BG-SQS. The common active time between each child node v and its parent node u is $r_u l_v$. Thus $r_u l_v = r_u l_w$ if v and w are child nodes of u. The v's listening time is $(R_u - r_u) l_v$. The energy consumption difference between two nodes v and w is caused by their transmission power P_t^v and P_t^w. Thus from Equation (6) we can obtain that a child node with higher transmission power would consume more energy in a same basic group. In other words, the child node, which has highest transmission power, has shortest lifetime in a same basic group.

In Equation (5), $t_l^u = (L_u - d_u l_v) r_u \tau = \frac{m}{R_u} r_u \tau - d_u l_v r_u \tau$. In Equation (6), $t_l^v = (R_u - r_u) l_v \tau = R_u l_v \tau - l_v r_u \tau$. We can adjust the listening time of both a parent u and its child nodes by adjusting R_u.

Now the question is how to solve the mix-max problem in Equation (3). In other words, we'd minimize the energy consumption of all nodes in a basic graph according to Equation (5) and (6). Notice the transmission power P_t of both a parent node and its child nodes can not be changed when the tree $\mathscr{T}$ is constructed previously. When each node is informed of the task, the demand of each node $\mathcal{D}_v$ and $\mathcal{D}_{uv}$ in Equation (6) is determined in each period. Thus the only way to decrease the energy consumption of each node is to adjust R, L, r and l in Equation (5) and (6). But we find that

Lemma 4.2. *When the numbers of the SQS rows and columns are fixed, each child node can minimize its energy consumption by choosing minimal number of NSQs.*

Proof. Suppose u is a parent node of two child node v and w. Notice that $(1+\zeta)\frac{\varphi_v}{\varrho}\mathcal{D}_v = l_v r_v \tau$. From Equation (6), we can obtain the energy consumption of two child nodes v and w.

$$\begin{aligned} E_v = P_t^v \frac{1}{1+\zeta} l_v r_v \tau + P_r \frac{\zeta}{1+\zeta} l_v r_v \tau \\ + P_l \tau (l_v R_u - l_v r_v) \end{aligned} \tag{7}$$

Where l_v is the number of NSQs selected by v and r_v is the number of rows in the l_v NSQs. In these rows, v has to finish its demand.

$$\begin{aligned} E_w = P_t^w \frac{1}{1+\zeta} l_w r_w \tau + P_r \frac{\zeta}{1+\zeta} l_w r_w \tau \\ + P_l \tau (l_w R_u - l_w r_w) \end{aligned} \tag{8}$$

Since the demand of a node is determinate in each period, $l_v r_v = (1+\zeta)\frac{\varphi_v}{\varrho\tau}\mathcal{D}_v$ is determinate and r_v changes with l_v. For example, suppose node w originally selects NSQ Q_x and it implements its demand in the time slots 1 and 7 of Q_x. But it changes its NQS and its Q_y and Q_z, and it can implement its demand in time slots 2 and 3 because it only needs two time slots.

In Equation (7) and (8), the listening power can be decreased by minimizing l_v and l_w when R_u and L_u are fixed. Thus it can minimize the energy consumption of the child nodes by minimizing the number of NSQs selected by child nodes. □

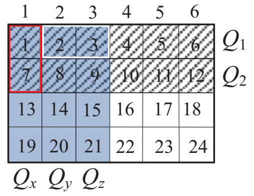


FIGURE 7. Each child node selects different amount of NSQs.

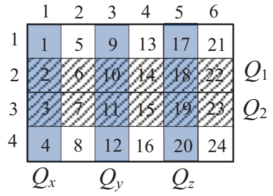


FIGURE 8. Quorum arrangement.

When each child node has same demand with other child node and sets more amount of NSQs than other child node, it would cost more energy according to Lemma 4.2. It also causes its parent to cost additional energy, for example, the parent node u has to cost additional time slots 8 and 9 on listening in Figure 7. Thus we have the followed lemma.

Lemma 4.3. *For any pair of child nodes in a same basic graph, they should select same amount of NQSs when they have same demand.*

From above lemma and Equation (7), we can know that the only reason that causes any pair of child nodes to consume different energy is the transmission power in a basic group. Thus we obtain the followed lemma.

Lemma 4.4. *In a basic graph, a child node with highest transmission power consumes maximal energy among all the child nodes when all the nodes have same demand.*

Notice $r_u = \max_{v \in c(u)} r_v$ in Equation (9) since the parent node should guarantee to have enough common active time with every child nodes. Lemma 4.3 indicates that a parent u should select $r_u = \max_{v \in c(u)} r_v = r_v$ BSQs in order to minimize the energy consumption. From Equation (5), we can obtain the energy consumption of the parent u.

$$\begin{aligned} E_u = P_t^u d_u \frac{\zeta}{1+\zeta} \tau l_v r_v + P_r d_u \frac{\tau}{1+\zeta} l_v r_v \\ + P_l \tau (\frac{m}{R_u} r_u - d_u l_v r_v) \\ = \tau l_v r_v d_u \{P_t^u \frac{\zeta}{1+\zeta} + \frac{P_r}{1+\zeta} + P_l (\frac{m r_u}{R_u d_u l_v r_v} - 1)\} \\ = \tau l_v r_v d_u \{P_t^u \frac{\zeta}{1+\zeta} + \frac{P_r}{1+\zeta} + P_l (\frac{m r_v}{R_u d_u l_v r_v} - 1)\} \\ = \tau l_v r_v d_u \{P_t^u \frac{\zeta}{1+\zeta} + \frac{P_r}{1+\zeta} + P_l (\frac{m}{R_u d_u l_v} - 1)\} \end{aligned} \quad (9)$$

In above equation, $l_v r_v$, P_t^u, P_r, P_l, ζ and d_u are fixed so there are two ways to decrease the energy consumption of the parent u: increasing l_v or R_u. Since increasing l_v contradicts with Lemma 4.2 and is equivalent to increasing R_u. Thus we decrease the energy consumption of a parent node by increasing R_u. But it would cause the increase of the child nodes' energy consumption to increase R_u according to Equation (7). So the problem is how to adjust R_u in order to minimize the energy consumption of a parent and its child, which has maximal transmission power according to Lemma 4.4.

We use Game Theory [21] to solve the problem in Equation (3). Game Theory is a powerful tool to model players' behaviors and their impact on the protocol performance in distributed method with self-interested players. According to Lemma 4.4 and the definition of our network lifetime, the lifetime is determined by the child node with the maximal energy consumption and the parent in a basic graph and the parent node. A parent node can decrease its energy consumption by increasing R_u according to Equation (9) but the energy consumption of its child node will increase in the same time according to Equation (7). Thus it is a game of time sharing between a parent and its child node with maximal transmission power in order to maximize the network lifetime. In the game of time sharing, there are two player: a parent, denoted by $\mathscr{P}_p$, and its child node with maximal transmission power, denoted by $\mathscr{P}_c$.

We firstly introduce some definitions in game theory. A game consists of a finite set of players $\mathbf{N} = \{1, 2, \cdots, N\}$. Each of the players $i \in \mathbf{N}$ selects a strategy $s_i \in \mathbf{s}_i$ with the objective of maximizing its utility u_i. The strategies of all players compose a strategy profile $\mathbf{S}$, *i.e.*, $\mathbf{S} = (\mathbf{s}_i)_{i \in \mathbf{N}}$. We denote by $\mathbf{S}_{-i}$ the collective strategies of all players except player i.

In the time sharing game of this paper, two players 1, denoted by $\mathscr{P}_p$, and 2, denoted by $\mathscr{P}_c$, share the same *strategic set*, and their *payoffs*, are respectively given by Equation (9) and (7). In this paper, the player $\mathscr{P}_p$ is the parent and the player $\mathscr{P}_c$ is its child with maximal transmission power in a basic graph. The strategy set of both $\mathscr{P}_p$ and $\mathscr{P}_c$ is $\mathbf{s}_1 = \mathbf{s}_2 = \{R\}$, where R is the number of the row in the SQS of $\mathscr{P}_p$. So $\mathbf{s}_1 = \mathbf{s}_2 = \{1, 2 \cdots, m-1\}$ and both of these two sets are finite. Since there are two players, their strategic set and payoffs in the game of time sharing, the game is called a *strategic-form (or normal) game*. Here the game is also called *finite game* because the strategic set is finite.

Definition 4.1. *Strategy $\boldsymbol{s} \in \boldsymbol{S}$ is a Nash equilibrium if $u_i(\boldsymbol{s}) \geq u_i(s_i, \boldsymbol{s}_{-i})$ $\forall s_i \in \boldsymbol{s}_i, \forall i \in \boldsymbol{N}$.*

The utility functions of $\mathscr{P}_p$ and $\mathscr{P}_c$ are monotone function of R according to Equation (9) and (7). Since the time sharing game is a finite game, we can obtain the followed lemma according to Theorem 11 of [22].

Theorem 4.1. *In the time sharing game, there is a Nash equilibrium.*

The reason why we look for the Nash equilibrium is that the equilibrium is the solution of Equation (3). Then the next question is how to find the equilibrium. From Equation (7), we can obtain the followed equation.

$$\begin{aligned} E_v = P_t^v \frac{1}{1+\zeta} l_v r_v \tau + P_r \frac{\zeta}{1+\zeta} l_v r_v \tau + P_l \tau (l_v R_u - l_v r_v) \\ = l_v r_v \tau \{P_t^v \frac{1}{1+\zeta} + P_r \frac{\zeta}{1+\zeta} + P_l (\frac{R_u}{r_v} - 1)\} \end{aligned} \quad (10)$$

Let 10 equal to Equation (9), we can have

$$\begin{aligned} & \tau l_v r_v d_u \{P_t^u \frac{\zeta}{1+\zeta} + \frac{P_r}{1+\zeta} + P_l (\frac{m}{R_u d_u l_v} - 1)\} \\ = \quad & l_v r_v \tau \{P_t^v \frac{1}{1+\zeta} + P_r \frac{\zeta}{1+\zeta} + P_l (\frac{R_u}{r_v} - 1)\} \\ \Rightarrow \quad & (d_u P_t^u \zeta - P_t^v) \frac{1}{P_l(1+\zeta)} + \frac{P_r (d_u - \zeta)}{P_l (1+\zeta)} + 1 - d_u \\ = \quad & \frac{R_u}{r_v} - \frac{m}{R_u l_v} \Rightarrow a = \frac{R_u}{r_v} - \frac{m}{R_u l_v} \end{aligned}$$

$$\Rightarrow \quad R_u^2 - aR_u - \frac{mr_v}{l_v} = 0$$
$$\Rightarrow \quad R_u = \frac{a}{2} \pm \sqrt{a^2 + \frac{4mr_v}{l_v}}$$
$$\Rightarrow \quad R_u = \frac{a}{2} + \sqrt{a^2 + \frac{4mr_v}{l_v}} \; (Because\ R_u > 0)$$
$$\Rightarrow \quad R_u = \frac{a}{2} + \sqrt{a^2 + \frac{4mr_u}{l_v}}$$
$$(Because\ r_u = \max_{v \in c(u)} r_v = r_v) \qquad (11)$$

Where $a = (d_u P_t^u \zeta - P_t^v)\frac{1}{P_l(1+\zeta)} + \frac{P_r(d_u-\zeta)}{P_l(1+\zeta)} + 1 - d_u$, and $r_u = r_v = (1+\zeta)\frac{\varphi_v}{\varrho l_v \tau}\mathcal{D}_v$. Since the quorum of each child node v has l_v and a parent u has d_u child nodes, the BG-SQS of u has at least $d_u l_v$ columns.

$$R_u = \min\{\lceil \frac{a}{2} + \sqrt{a^2 + \frac{4mr_v}{l_v}} \rceil, \frac{m}{d_u l_v}\}. \qquad (12)$$

Based on above analysis, we present the time scheduling in Algorithm 1, in which, each parent node firstly adjusts the row and column of its BG-SQS, then assigns the quorums to itself and its child nodes in its SQS. Recall that each parent node must be informed of the task to be implemented before assigning quorums and had known the IDs of its own and its child nodes after the tree $\mathscr{T}$ was established in **Step I**.

Algorithm 1 LOD

Input: The type ζ of the network task and IDs of a parent u and its child nodes.

Output: Each node obtains a quorum, *i.e.* the active time.

1: Each parent node u counts the number d_u of child nodes;
2: u collects the transmission power of all of its child nodes and selects the maximal transmission power P_t^v;
3: u assigns $l_v = 1$ NSQs for each child node v;
4: u selects r_u BSQs, where $r_u = (1+\zeta)\frac{\varphi_v}{\varrho l_v \tau}\mathcal{D}_v$;
5: u adjusts the row R_u of its BG-SQS according to Equation (12).

4.2. Effect of Asynchronization on LOD

In SQS, Lemma 3.3 guarantees the non-empty intersection between BSQs and NSQs when the network is asynchronous, which is discussed in Section 3. And the quorums should be arranged in order to avoid the intersection among NSQs because of asynchronization. For example, nodes v and w are respectively assigned Q_x and Q_y in Figure 7. And v actually is active at 2 when it sets itself to be active at time slot 1. Then v would conflict with w.

Sensor networks are often asynchronous because of imprecise clock and the nondeterminism in the latency caused mainly by Send time and Access time [23]. [23] tested the Berkeley Sensor Node [24] based on TinyOS and obtained the distribution of the receivers' phase offsets, which obeys Gaussian distribution with average $\mu = 0$ and deviation $\sigma = 11.1\mu sec$ under confidence 99.8%. In this paper, we denote the phase offset by δ and for example, we can let $\delta = 3\sigma$. Thus we can arrange the quorums including BSQs and NSQs as shown in Figure 8. The quorum arrangement is as follows. Since NSQs are assigned to different child nodes, there should be $f(\delta)$ (according to Equation (13)) unselected NSQs between the adjacent NSQs selected by different child as shown in Figure 8 when each child is assigned NSQs. For example, a child node v select the first l_v then the NSQs from $l_v + 1$ to $l_v + f(\delta)$ should not be assigned to any child node.

$$f(\delta) = \begin{cases} \lfloor \frac{\delta}{\tau R} \rfloor, & \frac{\delta}{\tau R} - \lfloor \frac{\delta}{\tau R} \rfloor < R_u - r_u, \quad (13) \\ \lceil \frac{\delta}{\tau R} \rceil, & otherwise. \quad (13') \end{cases}$$

Where u is a parent node in a basic graph. Notice that the demand, which can be implemented in each period and basic graph, would be decreased in order to satisfy Equation (13). Other way to satisfy Equation (13) is to increase the cardinality of each period, but it will increase the delay. When $\frac{r_v \tau + \delta}{\tau R} < 1$, the network delay and demand would not be decreased.

5. PROPERTIES OF LOD

In this section, we analyze the properties of LOD, presented in Algorithm 1. The concerned properties are channel utility, SQS load and the maximal demand.

The maximal demand. Each node u affords a demand $\mathcal{D}_u$ in each period. It is easy to know $\mathcal{D}_u \le \frac{\varrho m \tau}{\varphi_u}$ if the demand $\mathcal{D}_u$ is implementable and packet loss is considered. When the interference models $\mathcal{M}$ presented in the subsection 2.1 are considered, the demand $\mathcal{D}_u$ may be less. Notice in a same basic graph, nodes can not transmit or receive data at same time because the interference. Actually the wireless interference of one node interferes the communication in not only its own basic graph but those near it. Thus the demand in a basic graph should satisfy the condition in Lemma 5.1. Before we give out Lemma 5.1, we introduce a constant $c_3(\mathcal{M})$ which is determined by the interference model $\mathcal{M}$. We calculate $c_3(\mathcal{M})$ by the vertex coloring. The vertex coloring refers to coloring all basic graphs with minimal number of colors under the interference model $\mathcal{M}$. Thus under the interference model $\mathcal{M}$, if the nodes with the same color are active together, they are interference-free. Here, the constant $c_3(\mathcal{M})$ is the number of colors under $\mathcal{M}$.

In Section 4.2, there are $f(\delta)$ NSQs between adjacent NSQs selected by two child nodes while considering asynchronization. Thus $d_u(l_v + f(\delta)) \le L_u$ and there are at most $(L_u - d_u f(\delta))R_u\tau$ time to implement demands of all child nodes in each period.

Lemma 5.1. *In a basic graph, the demand of all nodes is at most* $\sum_{v \in c(u) \cup \{u\}} \mathcal{D}_v \leq \frac{\varrho(m - f(\delta) R_u)\tau}{\varphi_u c_3(\mathcal{M})}$, *where* u *is a parent node in a basic group.*

When $\frac{r_v \tau + \delta}{\tau R} < 1$ or the network works under (ultra) low duty cycle, $f(\delta) = 0$. So $\sum_{v \in c(u) \cup \{u\}} \mathcal{D}_v \leq \frac{\varrho m \tau}{\varphi_u c_3(\mathcal{M})}$. It means the asynchronization would have no effect on the maximal demand that each node can implement when $\frac{r_v \tau + \delta}{\tau R} < 1$ or the network works under (ultra) low duty cycle.

Channel Utilization. Channel utilization is a traditional metric to illustrate MAC protocols' efficiency [19]. [25] gave an equation to calculate the channel utilizations of B-MAC, sift, PTDMA and Z-MAC. We give the definition in Equation (14) to evaluate the efficiency of channel utilization under LOD.

$$\mathscr{U} = \frac{t_c}{t_c + t_l + t_s} \tag{14}$$

Where t_c is the communication time of a node including transmission and receiving. And t_l is the listening time of the node. And t_s is the sleeping time of the node, which means the node turns off its radio. In this paper, the channel is utilized in the common active time slots between BSQs and NSQs, which includes the time to transmit and receive. So the utilization under BG-SQS is described by the followed equation.

$$\mathscr{U} = \frac{l_v r_v}{c_3(\mathcal{M}) m} \tag{15}$$

Where l_v and r_v are determined by the demand of each node. So on one hand, the channel utilization can be adjusted by changing the cardinality T of the period. In other hand, it is also related to the demand in each period.

5.1. Period Arrangement

Suppose the whole lifetime of a network is composed by a serial of periods, which compose a period set $\mathbb{T} = \{T_1, T_2, \cdots\}$. Each period T_i contains m consecutive positive integer $i+1, i+2, \cdots, i+m$, where i is positive integer. Notice that each basic graph is assigned a subset of $\mathbb{T}$ but the periods should be carefully assigned among the basic graphs in the whole network in order to avoid the interference between the neighboring basic graphs. For example, in Figure 5, the node v is a child node of u and also the parent of nodes x and y. In other words, v locates in two basic graph and has two SQSs.

Recall the clock shift of a node u is denoted by $\delta_u(t)$. The clock shift $\delta_u(t)$ may cross more than one period, *i.e.* $\frac{\delta_u(t)}{m\tau} > 1$. For example, u and v are respectively assigned periods T_1 and T_2 in Figure 5. v also shares the period T_1 since v is child node of u. v will be active in two quorums respectively contained in T_1 and T_2 so v should be active in two different time slots set under synchronization. But the two quorums of v may intersect with each other when $\frac{\delta_u(t)}{m\tau} > 1$. Thus v should be assigned a period T_{1+j} when its parent u is assigned a period T_1, where j is a positive integer and $j \geq \lfloor \frac{\delta_u(t)}{m\tau} \rfloor$.

Therefore, the period arrangement problem in this paper is that when any pair of nodes u and v in the interference range of each other and the interference models are given in Section 2.1, the periods T_i and T_j respectively assigned to u and v should keep at least $\lfloor \frac{\delta_u(t)}{m\tau} \rfloor$ periods away, *i.e.* $|i - j| \geq \lfloor \frac{\delta_u(t)}{m\tau} \rfloor$. Under this requirement, it is a MinMax problem to assign each node a subset of $\mathbb{T}$.

5.2. Improvement of Energy Efficiency

Node must cost energy on the idle listening during in order to maintain its quorum. For example, the quorum Q_v has to wake up at the time slots: 5, 11, 17, 23, 29, 35, and from 25 to 30 in order to have non-empty intersection with the quorum Q_u as shown in Figure 10. Node u and v have only two time slots, 11 and 26, to communicate with each other if their clocks are synchronous. Thus, the energy efficiency is low. When the network would work under ultra low duty cycle, such as 1% or less, nodes would waste much energy on the idle listening, and the energy efficiency would be much low. This block presents methods to save the energy consumption on idle listening so as to improve the energy efficiency.

The first method is called QS Expansion, which expands the time slots of the quorums in the parent QS as child QS as shown in Figure 9. In this figure, the time slot 30 is expanded to a child QS. In the child QS, node can select an arbitrary quorum, such as Q_v, to wake up. Figure 9 shows only one generation expansion, and the child QS can also expand its own child QS. In this way, the time to wake up can be greatly reduced. According to the properties of the non-empty intersection and rotation closure, the quorums in the same QS has still non-empty intersection. After one generation of QS expansion, the saved time not to wake up in the example of Figure 9 can be easily calculated as follows. The size of both the parent and child QS is $\frac{2}{\sqrt{m}} - 1$, and then the saved time slots not to wake up is $(\frac{2}{\sqrt{m}} - 1)(m - \frac{2}{\sqrt{m}} + 1)$.

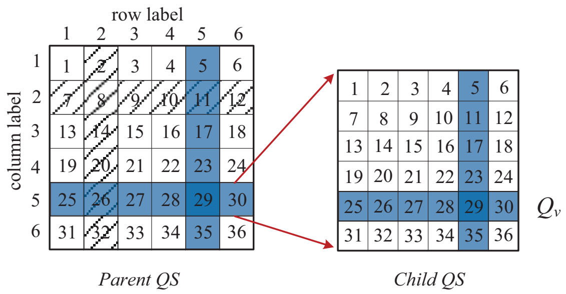


FIGURE 9. Extend a time slot as an quorum so that nodes can sleep in more time.

The second method is to shrink the idle listening time slots by estimating the clock shift. This method is quite simple and easy to implement. For example, the maximal clock shift is 10 seconds, and the size of each time slot is more than 10 seconds, such as 30 seconds. And then each quorum need only wake up at the intersected time slots, for example at the time slots 11 and 26 as shown in Figure 10. In this way, other nine time slots is saved not to wake up. This method can greatly save energy and increase the channel utilization.

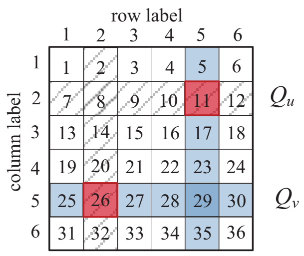


FIGURE 10. When the clock shift is much less than the size of a slot in a quorum, node can wake up during those intersected slots, such as 11 and 26.

Both of above methods can be directly applied to the SQS, designed in this paper.

6. EXPERIMENTAL RESULTS

This paper evaluates LOD and compares it with B-MAC in a real test-bed, which consists of 100 nodes and runs TinyOS 2.0 on TelosB nodes as shown Figure 11. We compare the performance of LOD against B-MAC on the network throughput, packet reception ratio (PRR) and energy consumption.

FIGURE 11. Our WSNs testbed with 120 nodes.

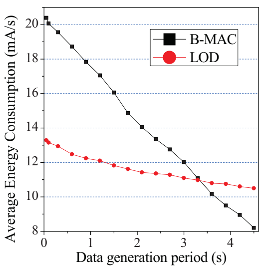


FIGURE 12. Energy Consumption Comparison between LOD and B-MAC.

6.1. Experimental Setup

On our test-bed, we start our experiment composed of two phases. At first stage, all nodes are initially set with 100% duty cycle and a clustered infrastructure, called a tree, is constructed by BFS. At the second phase, LOD and B-MAC are respectively implemented. Nodes in a same region selected their quorums according to their locations (parent or leaf node) in the tree and the number of the leaf nodes under LOD. Meanwhile, duty cycle is set as 20% in B-MAC.

Under LOD, each $\mathcal{Q}$ contains 100 time slots, *i.e.* $m = 100$. Each time slot is respectively set as $50ms$, $1s$, $2s$ and $5s$. Each node samples data in every $100ms$, $200ms$, $300ms$, $500ms$, $800ms$, $1s$, $1.5s$ and $2s$, which are called as the data generation period in Figure 13 and 14.

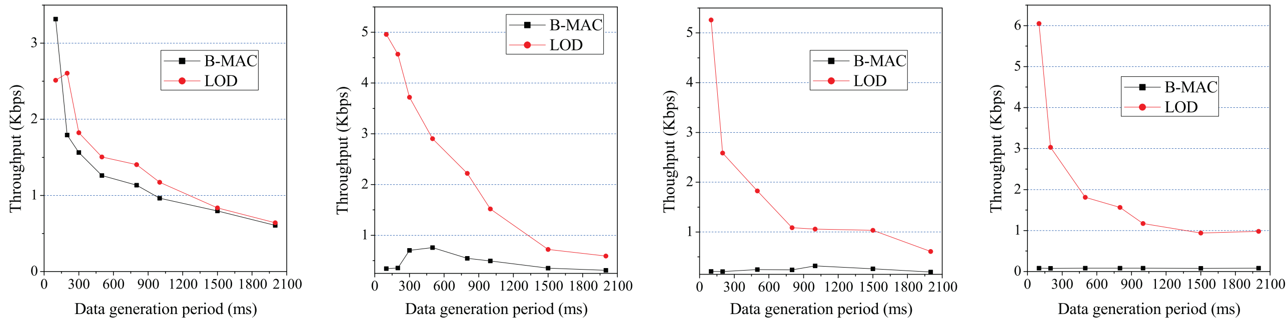
6.2. Performance Comparison

The evaluation of LOD and B-MAC on the network capacity and PRR illustrates their performance on channel utilization and fairness indirectly. Energy consumption is also measured.

Throughput. Figure 13 shows the network throughput under LOD and B-MAC respectively. Each node generates data at different rate. Because all nodes must compete channel access in B-MAC when transmitting packets, much channel resource is wasted. Thus B-MAC achieves lower network capacity than LOD, as shown in Figure 13. When sampling period is small, such as $100ms$, $200ms$ and $300ms$, the throughput under B-MAC is much lower than that under LOD. Because packets are transmitted more frequently under smaller sampling period, more contention for channel happens under B-MAC. Although the throughput under LOD is higher than B-MAC, it is actually quite low because the maximal data rate of TelsoB node can reach 250Kpbs. One of main reasons is that asynchronous clocks make quorums overlap with each other and clock shift of each node is arbitrary and isn't known in LOD. Thus connotative channel contention cannot be avoided. When the size of time slot is smaller, the possibility that quorums overlap is higher. Therefore, the throughput under both B-MAC and LOD is close to each other as shown in Figure 13(a).

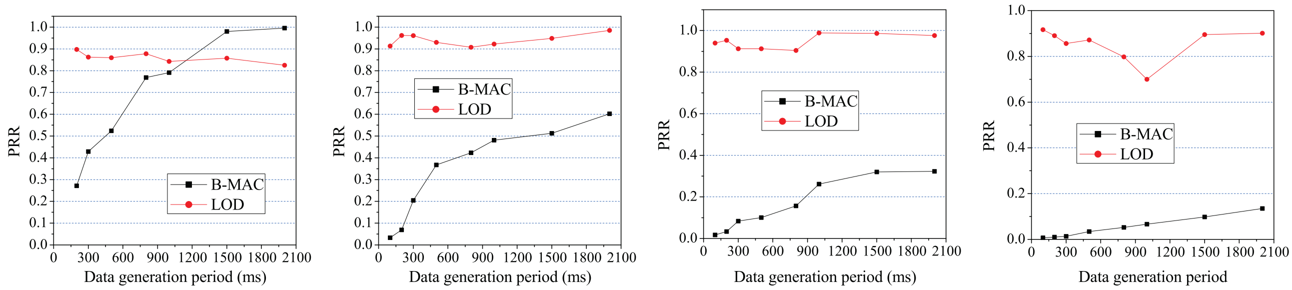
PRR. The PRR reflects the channel utility in the network. The PRRs under both B-MAC and LOD are shown in Figure 14. The PRR under both B-MAC increases with the increasing of the data generation period. But PRR under LOD almost keeps above 80%. The time slots size has much effect on the throughput under B-MAC instead of that under LOD. When the size of time slot is smaller, such as, $50ms$, PRR under LOD decreases much, and is lower than that under B-MAC when data generation is bigger than 900ms as shown in Figure 14(a). The reasons are similar to those on the network throughput.

Energy consumption. The experimental result on energy consumption is shown in Figure 12 when the size of time slot is 1s. In this figure, the energy consumption is the average value per node and per second. The energy consumption under LOD keeps about 6mA/s per nodes in different data



(a) Time slot size 50ms (b) Time slot size 1000ms (c) Time slot size 2000ms (d) Time slot size 5000ms

FIGURE 13. The network throughput respectively under LOD and B-MAC with different data generation periods.



(a) Time slot size 50ms (b) Time slot size 1000ms (c) Time slot size 2000ms (d) Time slot size 5000ms

FIGURE 14. The network PRR respectively under LOD and B-MAC with different data generation periods.

generation periods, while that under B-MAC varies from more than 20 mA/s to 8mA/s. It is obvious that energy consumption under B-MAC is greatly affected by the data generation periods because shorter data generation periods cause more contention for channel under B-MAC. The energy consumption in Figure 12 indicates that LOD can decrease the channel contention dramatically. Furthermore, we can notice that the energy consumption of B-MAC is lower than that of LOD when the data generation period is over 3.3 s. The reasons are twofold. Firstly, the channel contention decreases. Secondly, nodes lose more access to communicate with their neighbors because of the shift of their clocks.

7. RELATED WORK

7.1. Duty Cycle

In WSNs, operation at a certain duty cycle can bridge the gap between limited energy supplies and application lifetimes [26]. Gu and He designed a data forwarding technique to optimize the data delivery ration, end-to-end delay or energy consumption in low-duty-cycle sensor networks under a synchronized mode [1] . Guo and Gu *et al.* designed an Opportunistic Flooding scheme for low-duty-cycle networks with unreliable wireless links and predetermined wording schedules when the network is locally synchronized [15]. Wand and Liu provided a benchmark for assessing diverse duty-cycle-aware broadcast strategies and extend it to distributed implementation [27]. It translated the broadcast problem into a graph equivalence in order to seek a balance between efficiency and latency with coverage guarantees. Hong and Cao *et al.* proposed a set-cover-based approximation scheme with both centralized and distributed algorithms to minimized broadcast transmission delay [28].

Since the clocks are easy to shift in WSNs, some works considered the network operation under asynchronous clocks. The one of most famous techniques to adapt the asynchronization is the LPL technique [19]. Under this technique, some preambles is firstly sent before the true data so that nodes can communicate with some clock shift. The LPL technique cannot work well when the clock shift is bigger than the overall length of all the preambles. Sun and Johnson *et al.* designed an asynchronous duty-cycle broadcasting, under which, a node may be active very long time when it need broadcast the data a large number of neighbors, which awake up in different time [29].

There are also many other works studying the network performance under duty cycled networks, such as latency [30][31][32], opportunistic data aggregation [33], and reliable data delivery [34][35]. Nath and Gibbons analyzed the performance of geographic routing over duty-cycled nodes and presented a sleeping scheduling algorithm that can be tuned to achieve a target routing latency [36] and coverage [36]. After these works, researchers found that the network traffic or the demand has great affection on the duty cycle. Lee and Choi *et* *al.* proposed a traffic adaptive sensor network MAC: A-MAC to adjust the duty cycle of each node according to its traffic but it did not consider the synchronization among nodes and the feasibility of the traffic-based duty cycle adjustment since a network often affords of different kind of tasks and the traffic is variable in a fixed network [37].

7.2. MAC protocol

In WSNs, MAC is one main class of protocols considering the node sleep/wakeup or duty cycle adjustment. Their goals are to save energy, to improve the network throughput and/or to shorten the transmission delay. They can be classified into two classes: synchronization based and asynchronization based. Synchronization based protocols includes S-MAC [38], T-MAC [39], U-MAC [40], P-MAC [41] and H-MAC [42]. Asynchronization based ones includes D-MAC [43], B-MAC [19], Wise-MAC [44], SyncWUF [45] and ACDA [46]. Paper [46] gave the classification and the summarization of the above protocols.

Some protocols are designed to combine the advantages of TDMA and CSMA. Rhee and Warrier *et al.* proposed a hybrid MAC protocol, called Z-MAC [25]. In Z-MAC, a node always performs carrier-sensing before it transmits during a slot. Thus Z-MAC consumes much energy on the carrier-sensing. Z-MAC also needs local synchronization among senders in two-hop neighborhoods. S-MAC [38] and T-MAC [39] are also a hybrid of CSMA and TDMA and employ RTS/CTS mechanism to solve the the synchronization failure. Since these protocols use RTS/CTS, their overhead is quite high [25]. B-MAC [19] is the default MAC in the operate system of Mica2 and adopts Low Power Listening (LPL) to solve the asynchronization. Since LPL consumes much energy, X-MAC reduce the energy consumption and latency by employing short preamble and embedding address information of the target in the preamble [47]. Thus the non-target receivers can quickly go back to sleep and the energy is saved. LPL based preamble transmission may occupy the medium for much longer than actual data transmission. So [48] designed an asynchronous duty cycle MAC: RI-MAC. In RI-MAC, the energy would be wasted especially when the traffic load is low and the interference would be increased since each node should broad cast a beacon periodically no matter the sender has data to transmit or not.

MAC protocols are also designed to reduce energy consumption, such as S-MAC [38] and T-MAC [39]. Zheng and Hou *et al.* considered LPL approaches, such as WiseMAC and B-MAC, are limited to duty cycles of 1-2% and designed a new MAC protocol called scheduled channel polling (SCP) to ensure that duty cycles of 0.1% and below are possible [17]. It dynamically adjusts duty cycles in the face of busy networks and streaming traffic in order to reduce the latency.

Sha and Xing *et al.* presented a MAC protocol, C-MAC to achieve high-throughput bulk communication for data intensive sensing applications [49], but it did not consider duty cycle. Kim and Shin *et al.* proposed a lighted-weight channel hopping mechanism, thus avoiding redundant channel assignment by not allocating fixed channels to the nodes [50]. Synchronization was also implemented by initial time synchronization, error compensation and time slot assignment and retrieval. However, message was cost in this process and duty cycle was not considered in the protocol.

Bian *et al.* applied QS to establish channel control in dynamic spectrum access networks [6]. Wu *et al.* used it save power [7]. Chaporkar *et al.* maximized throughput in limited information multiparty MAC with QS [8]. Lai and Ravindran presented quorum-based duty-cycling schedule where nodes send out a beacon message at the beginning of wake-up slots [9].

MAC mainly allocates channel resources for these conflicting neighbors. Some of them considered the communication among nodes when their clocks were asynchronous. However, they still need synchronization protocols to cooperate the communication time, or add extra preamble to overcome the clock shift, such as LPL. This paper argues a new method to ensure the communication among nodes without using any synchronization protocols in case of imprecise clock.

8. CONCLUSION

In this paper, we designed a SQS based on QS. Based on SQS, we proposed a localized scheme, *LOD*, to adaptively adjust the duty cycle according to its demand in the clustered network such that each node can fairly use the channel. *LOD* combines the advantages of both TDMA and CSMA and doesn't need extra synchronization algorithms. When the network is asynchronous, *LOD* can still guarantee that any pair of neighboring nodes has a common active time to communicate with each other. *LOD* can increase the channel utilization by adjusting the cardinality of the period based on the demand of each node in the same basic graph. We established a multi-hop network by a testbed of 100 TelosB nodes and evaluated LOD. Compared to a TinyOS implementation of B-MAC, LOD significantly improves the performance such as network throughput and PRR.

There are few works waiting to finish in near future. In this paper, we discuss the case under which, the child nodes in the same basic graph have same amount of demand in each period. And the demand of a node is different from others' even in a same basic graph since the network may afford of variant kinds of tasks, such as routing. The challenge problem is how to schedule the active time for each node when nodes have different demands in each period. Furthermore, we discuss the application of SQS in tree-type infrastructure networks. SQS can be applied to other type networks in which, not every pair of neighboring nodes should have common time to communicate with each other in some application scenarios.

ACKNOWLEDGEMENT

We thank the anonymous reviewers for their valuable feedback helping to improve this paper. This work is supported by the National Natural Science Foundation of China (NSFC) under Grants No.61003298, the Major Program of the NSFC under Grant No. 61190113, the sub-foundation of Zhejiang Provincial Key Innovation Team on Sensor Networks under Grant No.2009R50046-7 and the open funding of Science and Technology on Sonar Laboratory in Hangzhou City under Grant No. KF201103.

REFERENCES

[1] Y. Gu and T. He. Data forwarding in extremely low duty-cycle sensor networks with unreliable communication links. In *ACM Proceedings of the 5th international conference on Embedded networked sensor systems (Sensys)*, pages 321–334, Sydney, Australia, Nov.6-9 2007.

[2] F. Sivrikaya and B. Yener. Time synchronization in sensor networks: a survey. *IEEE Network*, 18(4):45–50, 2004.

[3] Thomas Schmid, Zainul Charbiwala, Zafeiria Anagnostopoulou, Mani B. Srivastava, and Prabal Dutta. A case against routing-integrated time synchronization. In *ACM Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems (Sensys)*, pages 267–280, Zürich, Switzerland, Nov.3-5 2010.

[4] L. Mo, Y. He, Y. Liu, J. Zhao, S.J. Tang, X.Y. Li, and G. Dai. Canopy closure estimates with greenorbs: Sustainable sensing in the forest. In *ACM Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems (Sensys)*, pages 99–112, Berkeley, California, Nov.3-6 2009.

[5] D. Malkhi and M. Reiter. Byzantine quorum systems. *Springer Distributed Computing*, 11(4):203–213, 1998.

[6] K. Bian, J.M. Park, and R. Chen. A quorum-based framework for establishing control channels in dynamic spectrum access networks. In *ACM The Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 25–36, Beijing, China, Sept.20-25 2009.

[7] S.H. Wu, M.S. Chen, and C.M. Chen. Fully adaptive power saving protocols for ad hoc networks using the Hyper Quorum System. In *IEEE The 28th International Conference on Distributed Computing Systems (ICDCS)*, pages 785–792, Beijing, China, June 17-20 2008.

[8] P. Chaporkar, S. Sarkar, and R. Shetty. Dynamic quorum policy for maximizing throughput in limited information multiparty MAC. *IEEE/ACM Transactions on Networking (TON)*, 14(4):848, 2006.

[9] Shouwen Lai and Binoy Ravindran. On multihop broadcast over adaptively duty-cycled wireless sensor networks. In *The 6th IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS)*, pages 158–171, California, USA, June 21-23 2010.

[10] S. Li, Y. Liu, and X.Y. Li. Capacity of large scale wireless networks under Gaussian channel model. In *Proceedings of the 9th ACM Interational Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pages 140–151, Hongkong SAR, China, May 27-30 2008.

[11] Y.C. Tseng, C.S. Hsu, and T.Y. Hsieh. Power-saving protocols for IEEE 802.11-based multi-hop ad hoc networks. *Elsevier Computer Networks*, 43(3):317–337, 2003.

[12] M. Naor and A. Wool. The load, capacity, and availability of quorum systems. In *Annual Symposium on Foundations of Computer Science*, volume 35, pages 214–214, 1994.

[13] J.R. Jiang, Y.C. Tseng, C.S. Hsu, and T.H. Lai. Quorum-based asynchronous power-saving protocols for IEEE 802.11 ad hoc networks. *Springer Mobile Networks and Applications*, 10(1):169–181, 2005.

[14] P.J. Wan, Z. Wang, H. Du, S.C.H. Huang, and Z. Wan. First-fit scheduling for beaconing in multihop wireless networks. In *The 29th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 1–8, SanDiego, CA, USA, March 15-19 2010. IEEE.

[15] S. Guo, Y. Gu, B. Jiang, and T. He. Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links. In *ACM The Annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 133–144, Beijing, China, Sept.20-25 2009.

[16] L. Chen and J. Leneutre. A game theoretic framework of distributed power and rate control in IEEE 802.11 WLANs. *IEEE Journal on Selected Areas in Communications*, 26(7):1128–1137, 2008.

[17] R. Zheng, J.C. Hou, and L. Sha. Asynchronous wakeup for ad hoc networks. In *ACM MobiHoc*, pages 35–45. ACM, 2003.

[18] P. Dutta and D. Culler. Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications. In *ACM Proceedings of the 6th ACM Conference on Embedded Networked Sensor Systems (Sensys)*, pages 71–84, Raleigh, NC, USA, Nov.5-7 2008. ACM.

[19] J. Polastre, J. Hill, and D. Culler. Versatile low power media access for wireless sensor networks. In *ACM Proceedings of the 2th ACM Conference on Embedded Networked Sensor Systems (Sensys)*, pages 03–05, Baltimore, Maryland, Nov.3-5 2004.

[20] TelosB Datasheet. Crossbow Inc.

[21] M.J. Osborne and A. Rubinstein. *A course in game theory*. The MIT press, 1994.

[22] A.B. MacKenzie and L.A. DaSilva. Game theory for wireless engineers. *Synthesis Lectures on Communications*, 1(1):1–86, 2006.

[23] J. Elson, L. Girod, and D. Estrin. Fine-grained network time synchronization using reference broadcasts. *ACM SIGOPS Operating Systems Review*, 36(SI):147–163, 2002.

[24] J.M. Kahn, R.H. Katz, and K.S.J. Pister. Next century challenges: mobile networking for "Smart Dust". In

ACM/IEEE The Annual International Conference on Mobile Computing and Networking (MobiCom), pages 271–278, Washington, USA, August 15-20 1999.

[25] I. Rhee, A. Warrier, M. Aia, J. Min, and M.L. Sichitiu. Z-MAC: a hybrid MAC for wireless sensor networks. *IEEE/ACM Transactions on Networking (TON)*, 16(3):511–524, 2008.

[26] Jianhui Zhang, Shaojie Tang, Xingfa Shen, Guojun Dai, and Amiya Nayak. Quorum-based localized scheme for duty cycling in asynchronous sensor networks. In *IEEE 8th International Conference on Mobile Adhoc and Sensor Systems (MASS)*, pages 440–449, Valencia, Spain, October 17-22.

[27] F. Wang and J. Liu. Duty-cycle-aware broadcast in wireless sensor networks. In *The 28th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 468–476, Rio de Janeiro, Brazil, April 19-25 2009.

[28] J. Hong, J. Cao, W. Li, S. Lu, and D. Chen. Minimum-Transmission Broadcast in Uncoordinated Duty-Cycled Wireless Ad Hoc Networks. *IEEE Transactions on Vehicular Technology (TVT)*, 59(1):307–318, 2010.

[29] Y. Sun, O. Gurewitz, S. Du, L. Tang, and D.B. Johnson. ADB: an efficient multihop broadcast protocol based on asynchronous duty-cycling in wireless sensor networks. In *ACM Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems (Sensys)*, pages 43–56, Berkeley, California, Nov.4-6 2009.

[30] X. Jiao, W. Lou, J. Ma, J. Cao, X. Wang, and X. Zhou. Duty-cycle-aware minimum latency broadcast scheduling in multi-hop wireless networks. In *IEEE 30th International Conference on Distributed Computing Systems (ICDCS)*, pages 754–763, June 21-25, Genoa, Italy 2010.

[31] Shuyun Luo, Xufei Mao, Yongmei Sun, Yuefeng Ji, and Shaojie Tang. Delay minimum data collection in the low-duty-cycle wireless sensor networks. In *IEEE Global Communications Conference (GLOBECOM)*, pages 232–237, California,USA, Dec. 3-7 2012.

[32] Xiaodong Wang, Xiaorui Wang, Liu Liu, and Guoliang Xing. Dutycon: A dynamic duty-cycle control approach to end-to-end delay guarantees in wireless sensor networks. *ACM Transactions on Sensor Networks (TOSN)*, 9(4):42, 2013.

[33] J. Benson, T. O'Donovan, U. Roedig, and C.J. Sreenan. Opportunistic aggregation over duty cycled communications in wireless sensor networks. In *ACM/IEEE Proceedings of the 7th international conference on Information processing in sensor networks (IPSN)*, pages 307–318, Missouri, USA, April 22-24 2008.

[34] Giuseppe Anastasi, Eleonora Borgia, Marco Conti, and Enrico Gregori. A hybrid adaptive protocol for reliable data delivery in wsns with multiple mobile sinks. *Oxford Journal: The Computer Journal*, 54(2):213–229, 2011.

[35] Feng Wang and Jiangchuan Liu. On reliable broadcast in low duty-cycle wireless sensor networks. *IEEE Transactions on Mobile Computing (TOMC)*, 11(5):767–779, 2012.

[36] S. Nath and P.B. Gibbons. Communicating via fireflies: Geographic routing on duty-cycled sensors. In *ACM/IEEE Proceedings of the 6th international conference on Information processing in sensor networks (IPSN)*, pages 440–449, Massachusetts, USA, April 25-27 2007.

[37] S.H. Lee, J.H. Park, and L. Choi. AMAC: Traffic-Adaptive Sensor Network MAC Protocol through Variable Duty-Cycle Operations. In *Proceedings of IEEE International Conference on Communications (ICC)*, Scotland, 2007.

[38] W. Ye, J. Heidemann, and D. Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Transactions on Networking (TON)*, 12(3):506, 2004.

[39] T. Van Dam and K. Langendoen. An adaptive energy-efficient MAC protocol for wireless sensor networks. In *ACM Proceedings of the 1th ACM Conference on Embedded Networked Sensor Systems (Sensys)*, page 180, Los Angeles, CA, USA, Nov.5-7 2003.

[40] S.H. Yang, H.W. Tseng, E.H.K. Wu, and G.H. Chen. Utilization based duty cycle tuning MAC protocol for wireless sensor networks. In *IEEE Global Telecommunications Conference (GLOBECOM)*, page 5, St. Louis, MO, USA, Nov.28- Dec.2 2005.

[41] T. Zheng, S. Radhakrishnan, and V. Sarangan. PMAC: an adaptive energy-efficient MAC protocol for wireless sensor networks. In *19th IEEE Proceedings of International Parallel and Distributed Processing Symposium (IPDPS)*, page 8, 2005.

[42] H. Wang, X. Zhang, and A. Khokhar. An energy-efficient low-latency MAC protocol for wireless sensor networks. In *IEEE Global Telecommunications Conference (GLOBECOM)*, pages 1–5, San Frnacisco,Clifornia, USA, Nov.27-Dec.1 2006.

[43] G. Lu, B. Krishnamachari, and C.S. Raghavendra. An adaptive energy-efficient and low-latency MAC for data gathering in wireless sensor networks. In *Proceedings 18th International Parallel and Distributed Processing Symposium (IPDPS)*, page 224, Santa Fe, New Mexico, USA, April 26-30 2004.

[44] A. El-Hoiydi and J.D. Decotignie. WiseMAC: an ultra low power MAC protocol for the downlink of infrastructure wireless sensor networks. In *ISCC*, pages 244–251. IEEE, 2004.

[45] X. Shi and G. Stromberg. SyncWUF: An ultra low-power MAC protocol for wireless sensor networks. *IEEE Transactions on Mobile Computing (TOMC)*, pages 115–125, 2007.

[46] Y.C. Chang, J.R. Jiang, J.P. Sheu, and H.Y. Shih. ADCA: an asynchronous duty cycle adjustment MAC protocol for wireless sensor networks. In *IEEE Global Communications Conference (GLOBECOM)*, pages 1–5, New Orleans, LA, USA, No.30-De.4 2008.

[47] M. Buettner, G.V. Yee, E. Anderson, and R. Han. X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks. In *ACM Proceedings of the 4th ACM Conference on Embedded Networked Sensor Systems (Sensys)*, page 320, Boulder, Colorado, USA, Oct.31-Nov.3 2006.

[48] Y. Sun, O. Gurewitz, and D.B. Johnson. RI-MAC: a receiver-initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks. In *ACM Proceedings of the 6th ACM*

Conference on Embedded Networked Sensor Systems (Sensys), pages 1–14, Raleigh, North Carolina, USA, Raleigh, NC, USA 2008.

[49] M. Sha, G. Xing, G. Zhou, S. Liu, and X. Wang. C-mac: Model-driven concurrent medium access control for wireless sensor networks. In *The 28th Annual IEEE International Conference on Computer Communications (INFOCOM)*, pages 1845–1853, Rio de Janeiro, Brazil, April 19-25 2009.

[50] Y. Kim, H. Shin, and H. Cha. Y-mac: An energy-efficient multi-channel mac protocol for dense wireless sensor networks. In *ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pages 53–63, Missouri, USA, April 22-24 2008.