Energy-Efficient Quorum-Based MAC Protocol for Wireless Sensor Networks

L. Sherly Puspha Annabel and K. Murugan

The reliability of sensor networks is generally dependent on the battery power of the sensor nodes that it employs; hence it is crucial for the sensor nodes to efficiently use their battery resources. This research paper presents a method to increase the reliability of sensor nodes by constructing a connected dominating tree (CDT), which is a subnetwork of wireless sensor networks. It detects the minimum number of dominatees, dominators, forwarder sensor nodes, and aggregates, as well as transmitting data to the sink. A new medium access control (MAC) protocol, called Homogenous Quorum-Based Medium Access Control (HQMAC), is also introduced, which is an adaptive, homogenous, asynchronous quorum-based MAC protocol. In this protocol, certain sensor nodes belonging to a network will be allowed to tune their wake-up and sleep intervals, based on their own traffic load. A new quorum system, named BiQuorum, is used by HQMAC to provide a low duty cycle, low network sensibility, and a high number of rendezvous points when compared with other quorum systems such as grid and dygrid. Both the theoretical results and the simulation results proved that the proposed HQMAC (when applied to a CDT) facilitates low transmission latency, high delivery ratio, and low energy consumption, thus extending the lifetime of the network it serves.

Keywords: Wireless sensor networks, connected dominating set, wake-up scheduling, quorum, rotation closure, medium access control.

I. Introduction

A wireless sensor network (WSN) is a group of sensor nodes (arranged over a wide area) that have sensing, computing, and communicating capabilities. WSNs are broadly used in various applications such as military surveillance, habitat monitoring, inventory tracking, acoustic detection, pollution monitoring, medical systems, target tracking, robotic exploration, health care, environment monitoring, micro surgery, agriculture, and so on. In a WSN, a medium access control (MAC) protocol plays a very important role in energy conservation, since sensors are battery operated [1]. The prevailing power-saving protocols of today suffer from high latency and high energy consumption due to the non-adaptive nature of the varying traffic loads of networks. Therefore, a power-saving MAC protocol has to be developed such that it not only saves power but also guarantees successful transmission.

Scheduling is very important to reduce the energy consumption of a WSN by allowing the sensor nodes of the WSN to enter into a *sleep* state when not in use and a *wake-up* state when needed.

In a WSN, scheduling schemes are employed to assign time slots to the sensor nodes of the WSN. This helps to achieve reliable data communication using minimum energy, thereby prolonging the network lifetime of the WSN.

On-demand, synchronous, semi-synchronous, and asynchronous wake-up scheduling are the four important categories of existing wake-up scheduling schemes. In on-demand wake-up scheduling, sensor nodes are in a constant state of readiness and will wake when required to do so. In synchronous wake-up scheduling, all sensor nodes within the neighborhood of a specified cluster maintain a schedule to wake up and check for a medium with which to transmit data.

In semi-synchronous wake-up scheduling, sensor nodes periodically exchange their schedules with their immediate neighbors, synchronize with each other to form a cluster, and then interact among themselves in an asynchronous manner.

In asynchronous wake-up scheduling, clock synchronization is not required, and independent sleep and wake-up schedules will be taken up by each sensor node, thus reducing the message overhead [2].

A quorum system is a collection of sets or quorums in which any two sets have a nonempty intersection.

A WSN that employs a quorum system for the design of power-saving protocols assigns each sensor node of the network a specific cycle pattern with a guarantee that any two sensor nodes will intersect in at least one time slot within each clock cycle [3].

Quorum systems are classified into two categories; namely, homogenous and heterogenous. In homogenous quorum systems, all sensor nodes are of the same cycle length, but in heterogenous quorum systems, adjacent sensor nodes have different cycle lengths [4]. Traditional quorum systems are categorized as either grid, dygrid, tree, torus, extended torus, or adaptive quorum systems. In a grid quorum, the wake-up time intervals of sensor nodes will overlap only twice per duty cycle. In a dygrid quorum, sensor node wake-up times are scheduled adaptively and independently. However, the proposed BiQuorum system guarantees the intersection of such wake-up time intervals even when there is a clock drift.

Topology control allows sensor nodes to make decisions on their eligibility to join a network backbone. Furthermore, it helps in building reduced topology, thereby achieving energy saving. Topology construction is broadly classified into three types; namely, controlling transmission power, hybrid, and hierarchical. Hierarchical topology construction includes backbone-based construction, in which a connected dominating set (CDS) is largely used as the virtual backbone for a WSN. The main challenge of virtual backbone construction is to build a minimum CDS (MCDS).

To achieve minimum energy consumption, minimum end-to-end delay, and maximum packet delivery ratio, a new protocol called HQMAC is proposed that makes use of the aforementioned BiQuorum system. In BiQuorum, the quorum ratio, network sensibility (NS), and number of rendezvous points are improved in comparison with existing quorum systems. In this paper, a connected dominating tree (CDT), which includes the minimum number of dominators and connector sensor nodes, is constructed to improve the performance of a network [5]. The HQMAC protocol also adaptively changes the sleep and wake-up states of certain sensor nodes in the network it serves, according to the traffic load of the sensor node.

The rest of this paper is organized as follows. Research allied

to and motivation for this topic is discussed in Section II. The preliminaries and assumptions; CDT construction; BiQuorum system; asynchronous wake-up scheduling; and HQMAC protocol description are explained in Section III. The performance analysis of BiQuorum is discussed in Section IV. The results of our simulation are given in Section V, and the conclusion is specified in Section VI.

II. Literature Review

W. Ye and others developed the SMAC protocol, which is a synchronous MAC protocol. SMAC follows a fixed duty cycle; that is, sensor nodes alternate between sleep and wake-up states over a given time interval, which results in a delay in data transmission along with idle listening [6]. The Pattern-MAC, proposed by T. Zheng and others, is a MAC protocol whereby a sensor node can obtain patterns that inform it of activity in its immediate neighborhood [7]. If there is any activity within its immediate neighborhood, then the sensor node will enter into a wake-up state. An important drawback of this protocol is that if a sensor node does not correctly receive patterns, then the wake-up time intervals of sensor nodes may not intersect with each other, which would then result in idle listening and long transmission delays.

T.V. Dam and K. Langendoen have proposed the Timeout MAC (TMAC) protocol. In any WSN employing such a protocol, the sensor nodes will wait for a time-out period, T_A, and go into a sleep state if no activity occurs; hence, this will reduce idle listening [8]. TMAC suffers from a phenomenon known as the early sleeping problem, whereby potential *receivers* (sensor nodes in the receiving state) may go to sleep too early. C.M. Chao and Y.W. Lee defined the QMAC protocol, which is an asynchronous homogenous non-adaptive MAC protocol that has its own sleep and wake-up schedules and follows a fixed cycle length. This protocol uses a grid quorum and tries to extend network reliability by increasing the sleep times of nodes [9]–[11], but it does not support more than one channel for data transmission.

G. Ekbatanifard and others proposed a protocol called QueenMAC, which is an adaptive MAC protocol that makes use of a quorum system called dygrid [12]–[13]. In any WSN employing such a protocol, the sensor nodes will use dygrid to schedule their wake-up times based on the traffic load of the sensor node. QueenMAC outperforms the QMAC protocol due to the utilization of multiple channels for data transmission. Though multiple channels are used, as there is no flexible method for channel assignment, this protocol is vulnerable to collisions.

G. Lu and others have proposed a data-gathering MAC protocol, called the DMAC protocol, which attains low latency

[14]. The communication link between the various sensor nodes to the sink is embodied as the data-gathering tree, through which time slots are assigned to sensor nodes. A sensor node that follows the DMAC protocol has to wake up for every duty cycle. The major disadvantage of this protocol is that children of a particular sensor node will try to transmit data at the same wake-up time interval, which will result in collisions.

R. Yu and others have proposed an algorithm that consists of two phases for building a CDT [15]. In the first phase, the dominator identified as having the smallest ID is taken into consideration when forming a forest of sensor nodes, and in the second phase, groups of such dominators identified during the first phase are linked using connectors so as to form a CDS. The foremost drawback of this algorithm is that when a protocol is executed on the constructed CDS, it will indirectly consume more energy as the size of the CDS is not minimal.

Most existing quorum systems, such as the grid and dygrid quorum systems, have a high duty cycle, fewer rendezvous points, and are not adaptive to a high network traffic load [16]–[17]. In protocols such as the SMAC, Pattern-MAC, TMAC, and DMAC, the wake-up time intervals of sensor nodes may not intersect with each other, which leads to idle listening and long transmission delay. In QMAC, though the wake-up time intervals of sensor nodes are guaranteed to intersect with each other, this does not mean that the wake-up times are adaptively scheduled. Though QueenMAC adaptively changes the wake-up times of sensor nodes, based on the traffic load of the sensor node, it still suffers from high energy consumption because of the dygrid systems that it incorporates. Therefore, an energy-efficient MAC protocol has to be designed in such a way so that it satisfies the following constraints: (a) the wake-up time intervals of sensor nodes should be able to intersect in every cycle length, (b) the wake-up time intervals of sensor nodes must overlap even if there is a clock drift, and (c) a sensor node's sleep and wake-up frequencies are adjusted based on the traffic load of the sensor node network. The proposed HQMAC protocol is compared with the existing QueenMAC protocol and satisfies all of the above properties along with minimum quorum ratio, minimum NS, maximum number of rendezvous points, minimum energy consumption, end-to-end delay, and maximum packet delivery ratio.

III. HQMAC Protocol Description

1. Preliminaries and Assumptions

The assumptions made in the proposed algorithm are as follows: (a) all sensor nodes are static; hence, dynamic topology control for sensor nodes due to mobility is not taken into consideration, and (b) all sensor nodes have the same transmission range and send data to a common sink; this consists of two phases; namely, construction of a CDT and wake-up scheduling using the HQMAC protocol.

The first phase is to construct a CDT that consists of dominatees, dominators, connectors, and a sink sensor node or initiator sensor node. In the second phase, an energy-efficient MAC protocol called HQMAC is developed so as to let only the dominator sensor nodes go into sleep and wake-up states based on their own traffic load. This protocol makes use of a new grid system called BiQuorum, which helps the protocol to be more efficient in terms of energy and throughput. All dominators aggregate the data from their dominatees and forward them to the sink sensor node through connectors.

2. Construction of CDT

A wireless network is represented as a graph, $G(V, E)$, where the vertices are denoted by V and the edges are denoted by E. A subset S of V is said to be a dominating set (DS) if each and every sensor node in V of G is either available in the subset S or is adjacent to at least one sensor node in S. A CDS, which we will denote by C, of G is both a connected subgraph of G and a DS. Therefore, all sensor nodes in graph G can communicate to at least one sensor node in C with one hop distance. An independent set (IS) is a subset, which we will denote by D, of graph G if for any two vertices of D there is no edge between them. When any added sensor node to D violates the IS property of the graph, the IS is identified to be a maximal independent set (MIS), which is both an IS and a DS.

The construction of a CDT includes an initial phase, whereby an MIS is created to identify dominators. This is then followed by a second phase, whereby connectors are recognized and dominators are connected to the sink.

A. MIS Creation

The sensor nodes within an MIS, or the dominators, are the sensor nodes that act as the cluster head of a particular cluster. The coverage area of the cluster head is fixed. Any two sensor nodes within the coverage area will be considered as neighbor sensor nodes. The following algorithm, Algorithm 1, creates an MIS and identifies the dominatees.

Algorithm 1. Create MIS $(G(V, E))$.

1: //n: sensor node, N: neighbor, E_{res}: residual Energy, i: sink sensor node
2: Initialize MISBlack, NMISGrey = {φ}
3: initiatorNode = i
4: MISBlack = MISBlack U {i}
5: **for** all $n \in N(i)$ do
6:     NMISGrey = NMISGrey U {n}

```
7: endfor
8: do the following until all V ∈ (MISBlack ‖ NMISGrey)
9:    for all x∈N(i) do
10:      for all y∈N(x) do
11:        MaxRes(Y) = Max(E_res(y))
12:      if E_res(x) > MaxRes(Y)
13:        MISBlack = MISBlack U {x} //Dominators
14:          NMISGrey = NMISGrey U {N(x)} //Dominatees
15:      endif
16:      endfor
17:    endfor
```

Initially, all of the adjacent sensor nodes of the sink are colored grey. Among the neighbors of the grey sensor nodes, the sensor node with the highest energy is taken as the dominator, added to the set, and then changed to black; all the neighbors of the selected dominator are then colored grey. Therefore, as this process continues, we will eventually arrive at a situation where all of the dominators are colored black and all of the dominatees are colored grey. The energy estimation of dominator identification is done during each round, and a dominator will be selected in each round. The residual energy of the dominator identified in a given round j is calculated by subtracting the energy consumption of the dominator identified during the previous round, $j - 1$, from the residual energy of this $(j-1)$th-round dominator.

$$E_{rf}(j) = E_{rf}(j-1) - X_i(j-1), \qquad (1)$$

$$E_{rf}(j) = E_{rf}(j-1) - Y_i(j-1), \qquad (2)$$

where $X_i(j-1)$ is the energy consumption of the dominator X_i during round $j - 1$ and $Y_i(j-1)$ is the energy consumption of the cluster member Y_i during round $j - 1$. From (1) and (2), the residual energy of sensor node i during round j, $E_{rf}(j)$, is calculated. During each round, the sensor node with the highest residual energy; that is, $\max(E_{rf}(j))$, will be selected as the dominator for that round.

B. Creation of Connectors

The connectors are chosen in a distinctive way, by which the redundant dominators are not often used. After the dominator selection during each round, the one-hop neighbor sensor nodes of the sink that connects the maximum number of dominators turn out to be the connectors and are colored red. The sink, connectors, and dominators together form a CDS. The following algorithm, Algorithm 2, recognizes the relay sensor nodes or connectors required to connect the dominators to the sink sensor node.

Algorithm 2. Create Connectors $(G(V, E))$.

```
1: initiatorNode = i;
```

```
2: Initialize ConnectorRed, CDT = {ϕ}
3: for all n∈N(i) do
4:    find n that has maximum number of N(n) ∈ MISBlack
5:    ConnectorRed = ConnectorRed U {n} // Connectors
6:  endfor
7:  for all n ∈ MISBlack
8:    CDT = {CDT} U {n} U {ConnectorRed}U{NMISGrey}
9:  endfor
10: endwhile
```

If any cluster member or dominatee in the CDT needs to perform data transmission to the sink, then the dominator or the respective cluster head of the cluster transmits the data through the connector to the sink sensor node.

3. Wake-Up Scheduling

Once a CDT is constructed, the sleep and wake-up patterns will be chosen by each sensor node. The beacon time intervals that represent the sleep and wake-up states of the sensor nodes are classified into either quorum or non-quorum time intervals. Quorum time intervals signify the wake-up time, and the non-quorum time intervals signify the sleep time of the sensor nodes. The beacon time intervals of the sensor nodes are divided into 16 time slots.

A. BiQuorum

A quorum system is a superset of non-empty subsets of elements; each subset is called a quorum, which achieves the intersection property. In a quorum system, wake-up time intervals are organized in overlapping quorums, such that each quorum overlaps with every other quorum. Here, we propose the BiQuorum system — a system that is based on the quorum concept and that provides both an adaptive and low quorum ratio, which outperforms existing quorum systems. The term "Intersect" is used in the following definitions to denote the proposed quorum system.

Given a universal set $Z = \{0, 1, \ldots, n - 1\}$, where n is an integer, we denote X to be a set of non-empty subsets of Z, and it is termed to be an intersect if for all $A, A' \in X, A \bigcap A' \neq \{\phi\}$. For example, for a set $Z = \{0, 1, 2, 3, 4\}$, X is said to be an intersect if $X = \{\{1, 2\}, \{2, 3, 4\}, \{2, 4\}\}$. Here, any two subsets of X; that is, $\{1, 2\}\ldots\{2, 3, 4\}$ will have an intersection that will allow sensor nodes to meet with one another, where the elements of X are the wake-up time intervals of the sensor nodes.

Given a universal set $Z = \{0, 1, \ldots, n - 1\}$, where n is an integer, we denote A and B to be two non-empty subsets of Z. The pair (A, B) is termed to be a "BiIntersect" if for all $X\in A$, $Y\in B, X \bigcap Y \neq \{\phi\}$. For example, for a set $Z = \{0, 1, 2, 3, 4\}$, the
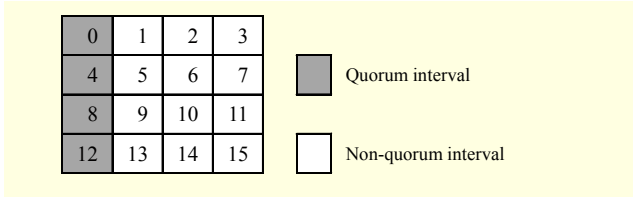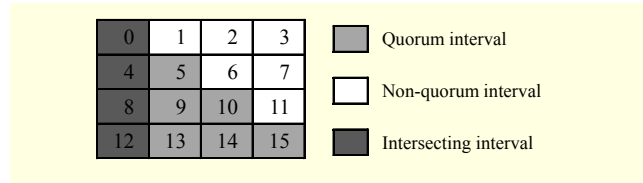
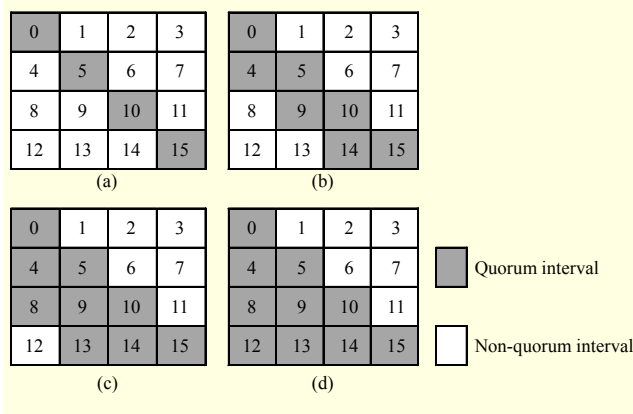Fig. 1. C-Intersect (1).

Fig. 3. (4, 1)BiQuorum.

Fig. 2. R-Intersect(X): (a) RI(1), (b) RI(2), (c) RI(3), and (d) RI(4).

pair (A, B) is said to be a BiIntersect if $A = \{\{0, 1\}, \{3, 4\}, \{0, 1, 4\}\}$ and $B = \{\{0, 3\}, \{0, 4\}\}$.

Definition 3.1: C-Intersect(Y). Given a universal set $Z = \{0, 1, \dots, n-1\}$, where n is a positive integer, we denote Y to be an integer whose value equals one. A "C-Intersect(1)," denoted as CI(1), is defined as

$$\text{CI}(1) = \left\{ \left(0, \sqrt{n}, 2\sqrt{n}, \dots, (\sqrt{n}-1)\sqrt{n}\right)(\bmod \; n)\right\}. \quad (3)$$

For example, in Fig. 1, when $n = 16$, and if n is shown as an $\sqrt{n} \times \sqrt{n}$ grid, CI(1) = {0, 4, 8, 12}.

Definition 3.2: R-Intersect(X). Given a universal set $Z = \{0, 1, \dots, n-1\}$, where n is a positive integer, we denote X to be an integer, $1 \leq X \leq \sqrt{n}$. An "R-Intersect(X)," denoted as RI(X), is defined as

$$\text{RI}(X) = \left\{ \left[(j-1)\left(1+\sqrt{n}\right)\right] \bmod \; n \right\}, \quad (4)$$

where, $i = 1, \dots, X$ and

$$j = \left(\left((i-1)\sqrt{n}\right)+1\right)\dots\left(\left(i\sqrt{n}\right)-(i-1)\right).$$

For example, if $n = 16$, and if n is shown as an $\sqrt{n} \times \sqrt{n}$ grid, when $i = 1, j = 1, 2, 3, 4$, then RI(4) = {0, 5, 10, 15}; and when $i = 2, j = 5, 6, 7$, then RI(4) = {4, 9, 14}; and when $i = 3, j = 9, 10$, then RI(4) = {8, 13}; and when $i = 4, j = 13$, then RI(4) = {12}. Therefore, RI(4) = {0, 4, 5, 8, 9, 10, 12, 13, 14, 15} since $X = 4$. Similarly, for RI(1) = {0, 5, 10, 15}, RI(2) = {0, 4, 5, 9, 10, 15}, and RI(3) = {0, 4, 5, 8, 9, 10, 13, 14, 15}; the

results are shown in Fig. 2.

Definition 3.3: ((X, 1)BiQuorum). Given an integer X, $1 \leq X \leq \sqrt{n}$, let C and D be the sets of the non-empty subsets of Z, where $Z = \{0, 1, 2, 3, \dots, n-1\}$. The pair (C, D) is called $(X, 1)$BiQuorum if and only if the following hold: (i) C is an RI(X) and D is a CI(1) or D is an RI(X) and C is a CI(1); and (ii) the pair (C, D) is a BiIntersect. For example, for $n = 16$, a $(4, 1)$BiQuorum is shown in Fig. 3.

The intersection between RI and CI is guaranteed. The number of intersections will range from 1 to $\sqrt{n}$ depending on the value of X. The advantage of this system is that the size of the RIs is substantially smaller when compared to those of the dygrid quorum system. For example, RI(4) and CI(1), which form the (4, 1)BiQuorum, have four intersections, as illustrated in Fig. 4. Similarly, $(\sqrt{n}, 1)$BiQuorum has $\sqrt{n}$ intersections.

B. Asynchronous Wake-Up Scheduling

In asynchronous wake-up scheduling, there is no need for time synchronization. In an HQMAC protocol, though the sensor nodes follow their own wake-up schedule, the overlapping among the wake-up intervals is guaranteed even if there is a clock drift.

Definition 3.4. For any given positive integer (a) and a quorum (Y) in a quorum system Q under $Z = \{0, \dots, n-1\}$, we define $\text{RC}(Y, a) = \{(a + b) \bmod n \mid b \in Y\}$.

Definition 3.5: Rotation closure property. A quorum system Q under $Z = \{0, \dots, n-1\}$ is said to have rotation closure if, for all $X, Y \in Q, a \in \{0, \dots, n-1\}: X \bigcap \text{RC}(Y, a) \neq \{\phi\}$.

Definition 3.6. The BiQuorum system fulfils the rotation closure property.

Proof. Let Q be a BiQuorum of size $\sqrt{n} \times \sqrt{n}$. Let $A \in Q$, which contains the left diagonal elements of the $\sqrt{n} \times \sqrt{n}$ grid; namely, $\{(0, \sqrt{n}+1, 2\sqrt{n}+2, \dots, n-1) \bmod n\}$. It then follows that $A \bigcap \text{RC}(B, a) \neq \{\phi\}$, where $B \in \text{CI}(1)$. Thus, the sensor nodes can tolerate time slot drifting, since the BiQuorum system fulfils the rotation closure. ∎

Theorem 1. Any hosts A and B that run the BiQuorum system have a maximum of X intersections for every n cycle intervals. Let $A \in \text{RI}(X)$ and $B \in \text{CI}(1)$. Then, for any two given integers, say a and b, we have $\text{RC}(A, a) \bigcap \text{RC}(B, b) \leq X$.

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | A |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | B |

Fig. 4. Overlapping of sensor node wake-up time intervals.

| 14 | 15 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | A |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | B |

Fig. 5. After two time-slot drifts in node *A*.

Proof. For RC(A, a) $\bigcap$ B, the maximum number of intersections when $X = 1$ is 1, and the maximum number of intersections when $X = 2$ is 2, and so on. Therefore, the total number of intersections is identified by

$$RC(A, a) \bigcap B \leq X. \tag{5}$$

Similarly, for $A \bigcap RC(B, b)$, the maximum number of intersections is 1 when $X = 1$, and the maximum number of intersections is 2 when $X = 2$, and so on. Therefore, the total number of intersections is identified by

$$A \bigcap RC(B, b) \leq X. \tag{6}$$

Thus, by (5) and (6), we have

$$RC(A, a) \bigcap RC(B, b) \leq X. \tag{7}$$

As can be seen in (7), it is certain that the two neighbor sensor nodes that adopt (X, 1)BiQuorum will meet each other within X time slots per n time slots, even if there is a clock drift.

If the sensor nodes A and B adopt RI(2) and CI(1) to form a (2, 1)BiQuorum as their pattern of the cycle, and if there is a time-slot drift equal to two, then the above theorem proves that both A's and B's wake-up time intervals will intersect at least twice within the cycle of 16. Figure 5 shows the overlapping of the time intervals in spite of two time-slot drifts in sensor node A. In RC(A, 2) $\bigcap$ RC(B, 0) of (2, 1)BiQuorum, there are two intersections.   ∎

C. HQMAC Protocol

The HQMAC protocol makes use of BiQuorum, whose important features, such as minimum quorum ratio, lead to high power saving and minimum NS; and the maximum number of rendezvous points leads to the maximum throughput and less data transmission delay. The wake-up times of dominators is determined by HQMAC, based on the traffic load of the sensor node. In addition, it also specifies to all remaining sensor nodes that there is no need for them to adjust their wake-up time intervals, except in the case of dominators.

The following algorithm, Algorithm 3, describes the HQMAC scheduling algorithm.

Algorithm 3. HQMAC.

1: //*i*: sink sensor node, *TD*: traffic load
2: **for** all $V \in$ G(V, E)
3:    **if** ($V == i$) **then** wake up = RI(4)
4:      **elseif** ($V \in$ NMISGrey || ConnectorRed) **then**
5:        wake up = CI(1)
6:      **endif**
7:    **endif**
8:    **while** ($V \in$ MISBlack)
9:      **if** ($TD >$ TH2) **then** wake up = RI(4)
10:        **elseif** (TH3 $< TD \leq$ TH2) **then** wake up = RI(3)
11:          **elseif** (TH4 $< TD \leq$ TH3) **then** wake up = RI(2)
12:            **elseif** (*TD* $\leq$ TH4*)* **then** wake up = RI(1)
13:            **endif**
14:          **endif**
15:        **endif**
16:      **endif**
17:    **endwhile**
18: **endfor**

The traffic load of a sensor node refers to the average number of transmitted data packets during one unit of time. During every round of cluster-head selection, the traffic load will be estimated. The latency is increased drastically when each sensor node's packet arrival time exceeds 300 kbps. Hence, the threshold of the traffic load (TH1) is assumed to be 300 kbps. Since the dominatees are sensor nodes that are situated away from the sink, only forward data, and are not comprehensively loaded, the wake-up times of these grey sensor nodes is taken as CI(1). Since the sink sensor node is heavily loaded with data from its connectors and adjacent sensor nodes, the wake-up time of this sensor node is taken as RI(4), which is the largest wake-up interval of RI. Since the dominatee sensor nodes are assumed to wake-up at CI(1), the connector sensor nodes are also set to the same wake-up state as that of the dominatee sensor nodes so as to obtain an

Dominatee	CI(1)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Dominator	RI(3)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Connector	CI(1)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Sink	RI(4)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Fig. 6. Dominator number of rendezvous points.

intersect with the time intervals of both the sink sensor node and the dominator sensor nodes. Therefore, the dominator sensor nodes are the sensor nodes that decide the intersecting time intervals of both the dominatees and the connectors. They either increase or decrease the intersecting time intervals of both the dominatee and the connector sensor nodes. Though energy-based dominator selection is used, the wake-up scheduling of a dominator is mainly based on the traffic load of the sensor node. This leads to minimum energy depletion, which in turn, results in infrequent dominator change.

As RI(4) follows 10 wake-up time intervals out of 16, the Threshold2 (TH2) is calculated as $300 \times (10/16) = 187.5$ kbps. Similarly, TH3 and TH4 values are calculated as 168.7 kbps and 131.2 kbps, respectively [18]. Therefore, when the traffic load (TD) is between TH3 and TH2, RI(3) will be assigned as the schedule of the dominator. Any sensor node having a TD greater than TH2 will be assigned with wake-up state RI(4), and if TD is less than TH4, then wake-up state RI(1) will be maintained by the sensor nodes.

Once the CDT is constructed, the data from the dominatee is transmitted in its intersecting wake-up intervals with the dominators; forwarded through the dominator; then forwarded to the connector; until finally, it reaches the sink sensor node. Figure 6 shows that when the dominator takes RI(3), the number of intersecting time intervals of the dominator with the dominatee and dominator with connector is three; that is, zero, four, and eight. Therefore, HQMAC specifies that only the dominators change their wake-up time intervals, while all other sensor nodes should remain in their predefined wake-up time intervals, which greatly reduces the overhead. HQMAC always allows the dominatee sensor nodes to be in the minimum wake-up state CI(1) = {0, 4, 8, 12}. During the other sleep states, a low power operating condition is maintained by the sensor nodes to sense the environment; subsequently, this does not allow them to deplete their energy. The dominators adopt a varying wake-up state based on the traffic load of the sensor node; thus, optimal energy depletion is achievable when sensor nodes take up both the proposed energy scheduling method and the traffic-aware scheduling method.

IV. Performance Analysis of BiQuorum

The most important metrics that are measured in the following experimental study with regard to the BiQuorum are
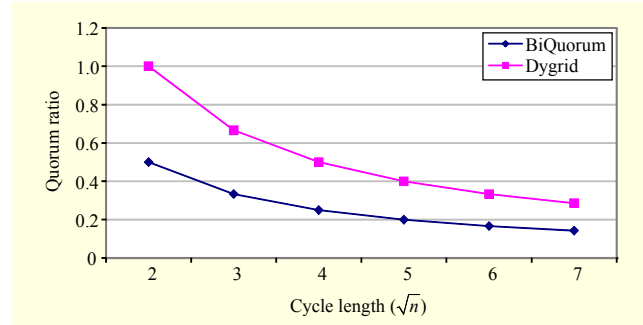

Fig. 7. Quorum ratio for different cycle lengths.

the quorum ratio, NS, and number of rendezvous points. The proposed system provides the maximum adaptive and minimum duty cycle and minimum NS when compared with the dygrid quorum system.

1. Quorum Ratio

A quorum ratio is defined to be the total number of time intervals required for a sensor node to be in a wake-up state in each cycle.

The major advantage of the BiQuorum is that the sizes of both the RI and the CI are at a minimum. Each sensor node will have a minimum of $\sqrt{n}$ duty cycles when compared to the dygrid quorum system, as illustrated in Fig. 7. The larger the size of the cycle lengths, the smaller is the quorum ratio.

2. NS

NS is the maximum delay for a sensor node to detect its neighbor sensor node. It is the maximum space between two farthermost common elements in the quorum system.

Theorem 2. Given a positive integer X, $X \leq \sqrt{n}$, the NS of $(X, 1)$, denoted as NS$(X, 1)$, is proved to be as follows:

$$\mathrm{NS}(X, 1) = \begin{cases} 2\sqrt{n} & X = 1, \\ 2\sqrt{n} - 1 & 1 < X \leq \sqrt{n}. \end{cases} \quad (8)$$

Proof. By Definition 3.2, any sensor node A that follows RI(X) will have a minimum of $\sqrt{n}$ elements whose NS$_A(X, 1)$ is identified as

$$\mathrm{NS}_A(X, 1) = \begin{cases} \sqrt{n} + 1 & X = 1, \\ \sqrt{n} & 1 < X \leq \sqrt{n}. \end{cases} \quad (9)$$
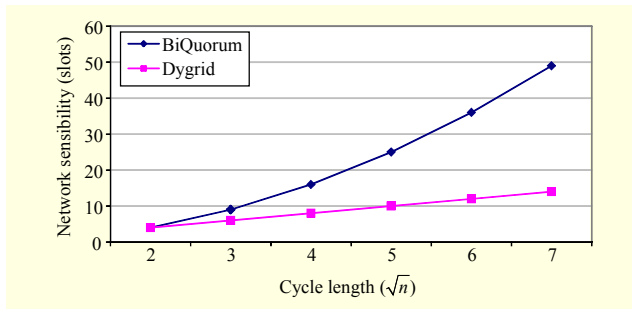
Fig. 8. NS for different cycle lengths.

By Definition 3.1, any sensor node B that follows CI(1) will have a maximum of $\sqrt{n}$ elements whose network sensibility NS_B is identified as

$$NS_B(X,1) = \sqrt{n} \quad \text{for } 1 \le X \le \sqrt{n}. \tag{10}$$

Therefore, from (9) and (10), NS(X, 1) is

$$NS(X, 1) = NS_A(X, 1) + NS_B(X, 1) - 1. \tag{11}$$

When $X = 1$,

$$NS(X, 1) = 2\sqrt{n}. \tag{12}$$

When $1 < X \le \sqrt{n}$,

$$NS(X, 1) = 2\sqrt{n} - 1. \tag{13}$$

Thus, from (12) and (13), NS(X, 1) is as follows:

$$NS(X, 1) = \begin{cases} 2\sqrt{n} & X = 1, \\ 2\sqrt{n} - 1 & 1 < X \le \sqrt{n}. \end{cases} \tag{14}$$

The maximum distance between any two successive elements for a sensor node that follows RI(X) is given in (9). The maximum distance between any two successive elements for a sensor node that follows CI(1) is given in (10). Therefore, the maximum distance between two successive common elements (that is, NS) is the sum of (9) and (10) minus one (the common element). Figure 8 shows a comparison of the NSs of the dygrid quorum system with the (1, 1)BiQuorum. It evidently shows that the BiQuorum system allows the sensor nodes to detect their neighbor sensor nodes with minimum delay. From Figs. 7 and 8, it is understood that the larger the quorum system's cycle length, the larger the NS and the smaller the quorum ratio. ∎

3. Rendezvous

Rendezvous is the total number of intersecting time intervals of any two sensor nodes. To provide two rendezvous points, a sensor node needs to be in a wake-up state for $2\sqrt{n}$ duty cycles in the dygrid quorum. In the BiQuorum system,

Table 1. Simulation parameters.

Simulation parameters	Value
Simulation time	50 s
Traffic source	CBR
Packet size	512 bytes
Transmit power	0.660 W
Receiving power	0.395 W
Idle power	0.035 W
Initial energy	10.1 J
Routing protocol	AODV
MAC	HQMAC, QueenMAC

to provide the same number of rendezvous points, the sensor node needs to be in a wake-up state for $2\sqrt{n} - 1$ time intervals.

V. Simulation Results

To evaluate the proposed HQMAC protocol's performance, an NS2 simulator is used for implementation. Table 1 displays the parameters used in the simulation. For a comparative case, the HQMAC protocol is assessed along with the MAC protocol QueenMAC; this is in addition to the evaluation of the proposed architecture framework with other sets of constraints. Each sensor node generates a 512 byte data packet every second. An initial energy of 10.1 J is given for every sensor node. The power consumptions for when the sensor nodes are in transmit mode, receive mode, and idle mode are 0.660 W, 0.395 W, and 0.035 W, respectively. The metrics used to assess the efficiency of the HQMAC are energy consumption, delivery ratio, and end-to-end delay.

1. Energy Consumption

The higher the energy consumption, the lower the network reliability. In Figs. 9, 10, and 11, the HQMAC protocol shows substantial decreases in energy consumption levels for various numbers of flows, sensor nodes, and arrival rates compared to the supplementary prevailing QueenMAC protocol. In QueenMAC, the sensor node duty cycle is increased as the traffic load is increased; however, energy consumption increases slowly, since sensor nodes use multiple channels to transmit data. The results clearly show that along with the benefits of the BiQuorum, which is incorporated in the HQMAC, the sensor node's duty cycle is adjusted dynamically. In the case of a high traffic load, an increase in intersecting time intervals (that is, a high rendezvous ratio) of the respective energy-efficient wake-up and sleep schedules saves energy.
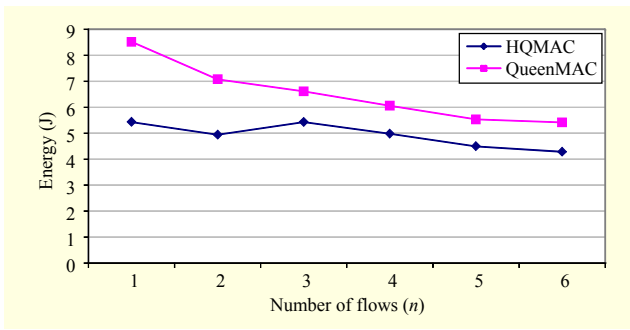
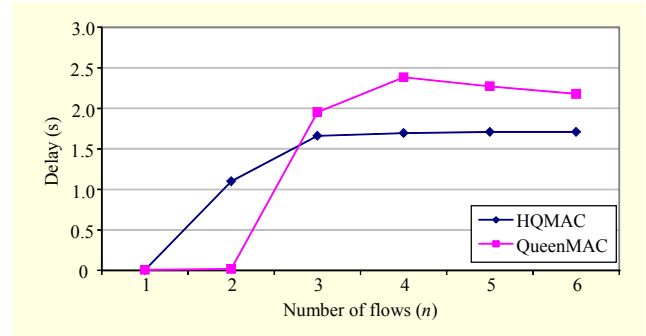Fig. 9. Energy consumption for various numbers of flows.
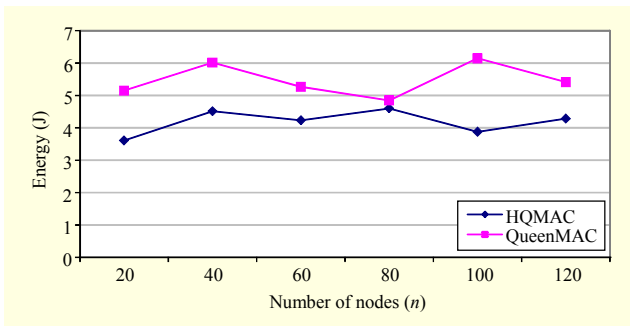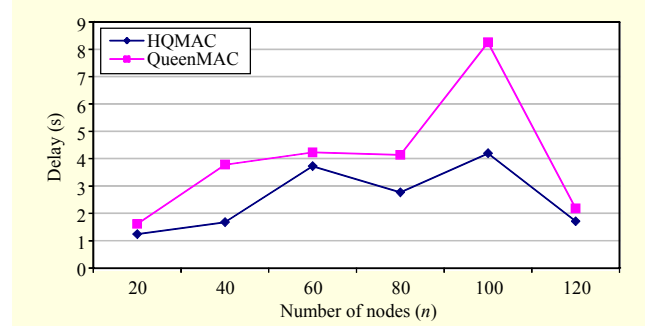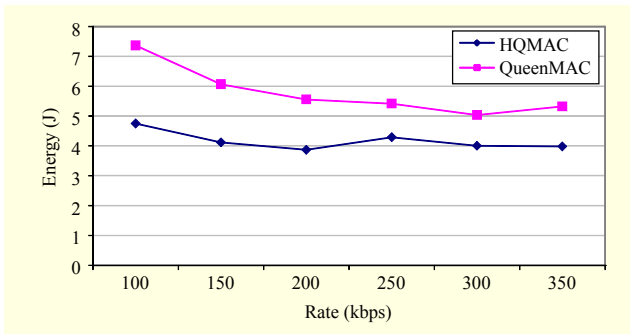

Fig. 10. Energy consumption for various numbers of sensor nodes.

Fig. 11. Energy consumption for various numbers of arrival rates.

Fig. 12. End-to-end delay for various numbers of flows.

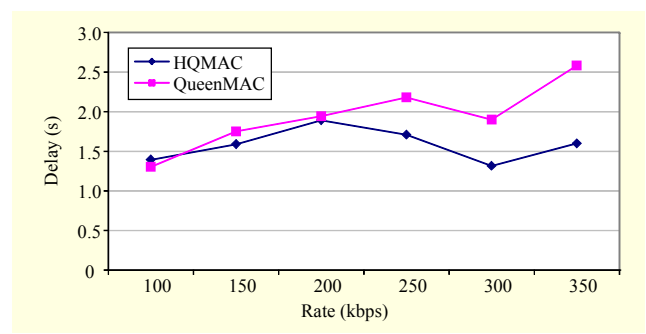Fig. 13. End-to-end delay for various numbers of sensor nodes.

Fig. 14. End-to-end delay for various numbers of arrival rates.

Though the HQMAC protocol provides a fixed duty cycle, the dynamic duty cycle adjustment based on the traffic load influences the energy consumption of the sensor nodes. Therefore, the consumption of energy is high when the traffic load is high; and vice versa.

2. End-to-End Delay

End-to-end delay is nothing but the time taken by the packets to reach the sink sensor node. Figures 12, 13, and 14 show the end-to-end delays of different MAC protocols for various numbers of flows, sensor nodes, and arrival rates, respectively. The results show that the delay of the HQMAC is lower than that of the QueenMAC. In QueenMAC, data transmission occurs from one sensor node to a greater number of sensor nodes, which escalates the forwarder sensor nodes and increases the collisions and number of retransmissions of data, thus leading to transmission latency. The maximum number of rendezvous points of the BiQuorum system aids the sensor nodes to meet each other and quickens the data transmission. Subsequently, the delay in a sensor node identifying its neighbor sensor nodes is minimized; thus, the transmission delay is also reduced.

3. Delivery Ratio

Delivery ratio is defined as the ratio of the amount of packets received successfully by the receiver to the amount of packets
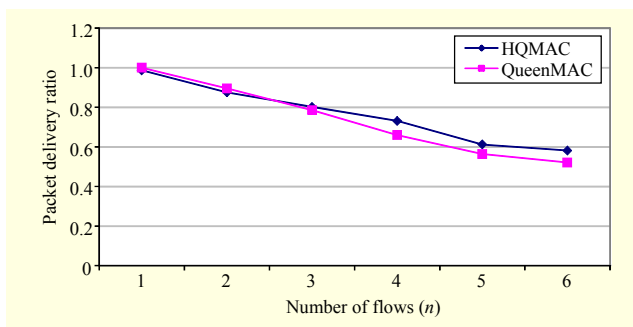
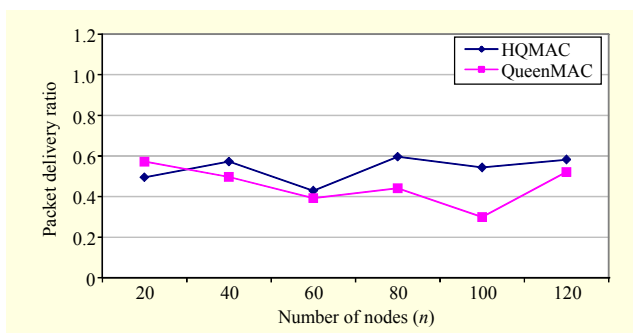Fig. 15. Packet delivery ratio for various numbers of flows.

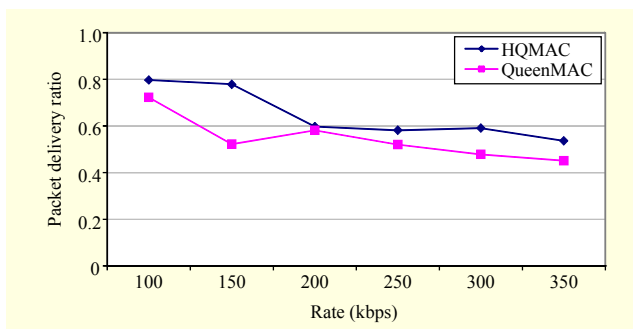Fig. 16. Packet delivery ratio for various numbers of sensor nodes.

Fig. 17. Packet delivery ratio for various numbers of arrival rates.

sent by the sender. The delivery ratio of different MAC protocols for various numbers of flows, sensor nodes, and arrival rates is shown in Figs. 15, 16, and 17, respectively. Though the QueenMAC protocol uses multiple channels for data transmission, it does not change its duty cycle depending upon the battery life of the sensor nodes; hence, it shows a reduction in its delivery ratio. In the HQMAC, the delivery ratio is increased because of the large number of intersecting quorum intervals within the low duty cycle and high adaptability when the traffic load varies.

VI. Conclusion

In this paper, a new asynchronous MAC protocol for WSNs called HQMAC is proposed, which adaptively tunes the wake-up and sleep intervals of sensor nodes, depending on the traffic load. To guarantee the overlapping of the wake-up time intervals of two adjacent sensor nodes, a new quorum system called BiQuorum is proposed. From the results, it is shown that BiQuorum delivers both a maximal quorum ratio and the maximum number of rendezvous points, as well as minimum NS when compared to the existing dygrid quorum system. To achieve network lifetime maximization, the HQMAC protocol is applied to a CDT that effectively determines the overall wake-up period of the dominatees, connectors, and sink sensor node in each cycle. It is clearly evident that the HQMAC protocol achieves high energy conservation by adaptively letting only the dominator sensor nodes enter into a wake-up state. The simulation results also show that the HQMAC protocol provides a low latency and high packet delivery ratio when compared to the QueenMAC protocol.

Acknowledgements

References

[1] I.F. Akyildiz et al., "Wireless Sensor Networks: A Survey," *Comput. Netw.*, vol. 38, no. 4, Mar. 2002, pp. 393–422.

[2] A. Keshavarzian, H. Lee, and L. Venkatraman, "Wakeup Scheduling in Wireless Sensor Networks," *ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, Florence, Italy, May 22–25, 2006, pp. 322–333.

[3] J.-R. Jiang, "Expected Quorum Overlap Sizes of Quorum Systems for Asynchronous Power-Saving in Mobile Ad Hoc Networks," *Comput. Netw.*, vol. 52, no. 17, Sept. 2008, pp. 3296–3306.

[4] S. Lai, B. Ravindran, and H. Cho, "Heterogenous Quorum-Based Wake-Up Scheduling in Wireless Sensor Networks," *IEEE Trans. Comput.*, vol. 59, no. 11, Nov. 2010, pp. 1562–1575.

[5] C. Zheng, L. Yin, and S. Sun, "Construction of d-Hop Connected Dominating Sets in Wireless Sensor Networks," *Procedia Eng.*, vol. 15, 2011, pp. 3416–3420.

[6] W. Ye, J. Heidemann, and D. Estrin, "Medium Access Control with Coordinated Adaptive Sleeping for Wireless Sensor Networks," *IEEE/ACM Trans. Netw.*, vol. 12, no. 3, June 2004, pp. 493–506.

[7] T. Zheng, S. Radhakrishnan, and V. Sarangan, "PMAC: An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks," *IEEE Int. Parallel Distrib. Process. Symp.*, Denver, CO, USA, Apr. 3–8, 2005, pp. 65–72.

[8] T.V. Dam and K. Langendoen, "An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks," *Proc. ACM SenSys*, Los Angeles, CA, USA, Nov. 5–7, 2003, pp. 171–180.

[9] C.-M. Chao and Y.-W. Lee, "A Quorum-Based Energy-Saving MAC Protocol Design for Wireless Sensor Networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 2, Feb. 2010, pp. 813–822.

[10] C.-M. Chao and Y.-W. Lee, "Quorum-Based Energy Saving MAC Protocol Design for Wireless Sensor Networks," *IEEE Int. Conf. Embedded Ubiquitous Comput.*, Shanghai, China, Dec. 17–20, 2008. pp. 316–322.

[11] L.S.P. Annabel and K. Murugan, "An Energy Efficient Wakeup Schedule and Power Management Algorithm for Wireless Sensor Networks," *Int. Conf. Recent Trends Inf. Technol.*, Chennai, India, Apr. 19–21, 2012, pp. 314–319.

[12] G. Ekbatanifard et al., "Queen-MAC: A Quorum-Based Energy-Efficient Medium Access Control Protocol for Wireless Sensor Networks," *Comput. Netw.*, vol. 56, no. 8, May 2012, pp. 2221–2236.

[13] G. Ekbatanifard and R. Monsefi, "MAMAC: A Multi-channel Asynchronous MAC Protocol for Wireless Sensor Networks," *IEEE Int. Conf. Broadband Wireless Comput.*, *Commun. Appl.*, Barcelona, Spain, Oct. 26–28, 2011, pp. 91–98.

[14] G. Lu, B. Krishnamachari, and C.S. Raghavendra, "An Adaptive Energy-Efficient and Low-Latency MAC for Tree-Based Data Gathering in Sensor Networks," *Wireless Commun. Mobile Comput.*, vol. 7, no. 7, Sept. 2007, pp. 863–875.

[15] R. Yu, X. Wang, and S.K. Das, "EEDTC: Energy-Efficient Dominating Tree Construction in Multi-hop Wireless Networks," *Pervasive Mobile Comput.*, vol. 5, no. 4, Aug. 2009, pp. 318–333.

[16] T.H. Hsu et al., "A Dynamic Traffic-Aware Duty Cycle Adjustment MAC Protocol for Energy Conserving in Wireless Sensor Networks," *Int. J. Distrib. Sensor Netw.*, 2012, pp. 1–10.

[17] Z.-T. Chou, Y.-H. Lin, and R.-H. Jan, "Optimal Asymmetric and Maximized Adaptive Power Management Protocols for Clustered Ad Hoc Wireless Networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 22, no. 12, Dec. 2011, pp. 1961–1968.

[18] C.-M. Chao, J.-P. Sheu, and I.-C. Chou, "An Adaptive Quorum-Based Energy Conserving Protocol for IEEE 802.11 Ad Hoc Networks," *IEEE Trans. Mobile Comput.*, vol. 5, no. 5, May 2006, pp. 560–570.

L. Sherly Puspha Annabel received her BE degree in computer science and engineering from the Karunya Institute of Technology, Coimbatore, India, in 2000 and her ME degree in computer science and engineering from Jaya Engineering College, Chennai, India, in 2006. She is currently performing research in the area of wireless sensor networks at Ramanujan Computing Centre, Anna University, Chennai, India. Her areas of interest include wireless ad hoc networking, sensor networks, and energy-efficient MAC protocols.

K. Murugan received his PhD degree in information and communication engineering from Anna University, Chennai, India, in 2006. He received his ME degree in computer science from the National Institute of Technology (Formerly REC), Tiruchirapalli, India, in 1992. He is currently working as a professor at Ramanujan Computing Centre, Anna University. He is a life member of ISTE, IETE, and CSI. His areas of interest include mobile computing, MANETs, and wireless sensor networks.