Computer Networks 56 (2012) 2221–2236

Contents lists available at SciVerse ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Queen-MAC: A quorum-based energy-efficient medium access control protocol for wireless sensor networks

GholamHossein Ekbatanifard *, Reza Monsefi, Mohammad H. Yaghmaee M., Seyed Amin Hosseini S.

Department of Computer Engineering, Ferdowsi University of Mashhad, Mashhad, Khorasan Razavi, Iran

ARTICLE INFO

Article history:
Received 2 August 2011
Received in revised form 29 December 2011
Accepted 11 March 2012
Available online 17 March 2012

Keywords:
Wireless Sensor Networks (WSNs)
Medium Access Control (MAC)
Data collection
Dyadic grid quorum systems (dygrid)
Energy-efficient

ABSTRACT

Major problems in the Medium Access Control (MAC) of Wireless Sensor Networks (WSNs) are: sleep/wake-up scheduling and its overhead, idle listening, collision, and the energy used for retransmission of collided packets. This paper focuses on these problems and proposes an adaptive quorum-based MAC protocol, Queen-MAC. This protocol independently and adaptively schedules nodes wake-up times, decreases idle listening and collisions, increases network throughput, and extends network lifetime. Queen-MAC is highly suitable for data collection applications. A new quorum system, dygrid is proposed that can provide a low duty cycle, $O(1/\sqrt{n})$, for adjusting wake-up times of sensor nodes. Theoretical analysis demonstrates the feasibility of dygrid and its superiority over two commonly used quorum systems (i.e., grid and e-torus). A lightweight channel assignment method is also proposed to reduce collision and make concurrent transmissions possible. Simulation results indicate that Queen-MAC prolongs the network lifetime while increasing the average delivery ratio and keeping the transmission latency low.

1. Introduction

Wireless Sensor Networks (WSNs) [1] have recently received much attention worldwide from military, industry, medical and health, urban traffic monitoring, and academia. WSNs are composed of many sensor nodes, each capable of gathering, processing, storing, and transferring environmental information. These nodes are usually organized in an ad hoc fashion. They operate in a distributed manner and coordinate with each other to accomplish a common duty.

The protocols designed for WSNs greatly depend on the applications for which the network has been established. Nonetheless, in many applications, one of the more serious challenges is how to increase the network lifetime now limited by the energy restriction of sensor nodes. Several factors are involved in the energy loss of nodes: collisions, retransmissions, idle listening, overhearing, and protocol overhead. The radio of a sensor node uses more power. The node usually turns its radio off, goes to sleep mode to save energy, and wakes up according to its predetermined schedule to transmit data. This method is called a duty cycling or a sleep scheduling [2], which is widely proposed for use in the Medium Access Control (MAC) protocol of multi-hop networks [3–8]. Different modes of a sensor node are shown in Fig. 1.

If designed properly, a MAC protocol can result in low power consumption and consequently increase the network lifetime. Most MAC protocols proposed for WSNs are based on the use of a single channel [8–17]. Such MAC protocols, especially in high-density deployments, increase collisions as well as end-to-end delay, and ultimately reduce the network lifetime. Several multi-channel MAC protocols [18–30] have been proposed recently with various objectives, e.g., handling burst traffic, fairness, reliability in data collection, evading external interference, improving throughput, and end-to-end delay. However,

* Corresponding author.
E-mail address: ekbatanifard@stu-mail.um.ac.ir (G. Ekbatanifard).

1389-1286/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.comnet.2012.03.004

energy saving is still an important issue. Existing energy saving mechanisms can be categorized into three types: synchronous, asynchronous, and on-demand wake-up [2].

Synchronous MAC protocols [18–20,22–27] normally maintain a schedule that specifies when a node should be awake to check the medium. These protocols cannot adapt to an individual's traffic well. In asynchronous MAC protocols [21,28,29] nodes independently schedule their wake-up times to periodically check the channel. When a node has data to send, it transmits a preamble that is long enough to be detected by the destination node. After preamble detection, the destination node stays awake to get the data following the preamble. These asynchronous protocols avoid the synchronization overhead. However, the long preamble results in longer latency and more energy consumption [31]. For on-demand wake-up protocols [30], nodes are equipped with a secondary low power radio to wake up its main radio to be ready for data exchange. Using multiple radio transceivers has some shortcomings. Radio transceivers consume energy, even while asleep, so increase the energy consumption of the nodes. In addition, a multiple radio transceivers system needs higher performance communication mechanisms and processor capabilities to receive and process data (or signals) from multiple channels.

Quorum systems recently have been utilized to design protocols for wireless networks [7,8,32–36]. There are several kinds of quorums: grid-based [37], torus [38], extended torus (e-torus) [32], and so on [35]. Some of them, such as grid and torus, have fixed duty cycles that makes them inappropriate for use in a network with different traffic conditions. The others such as e-torus, which has an adaptive duty cycle, provide a high minimum duty cycle that leads to more energy consumption if used in a network with a low traffic load.

This paper proposes a new quorum system, "dygrid". It surpasses existing quorums such as grid-based quorums [6,8,39], and e-torus [32], in terms of duty cycle, the number of rendezvous points, and network sensibility as discussed in Section 3.2. Utilizing adaptive dygrid, our proposed MAC protocol named "Queen-MAC" can save more energy while keeping the transmission latency low. For more energy saving, we also propose a lightweight channel assignment method to reduce collision and increase network throughput. Moreover, adaptive matching of wake-up intervals in Queen-MAC makes it flexible in different traffic conditions. Both theoretical analysis and simulation results are given to evaluate the performance of Queen-MAC in comparison with existing quorum-based [8] and multi-channel [28] MAC protocols. Theoretical analysis results demonstrate that the proposed protocol is more energy efficient while providing better network latency. Simulation results using OPNET Modeler 14.0 [40] verify that Queen-MAC increases the network lifetime and reduces network latency. The results also show that the performance of Queen-MAC is more significant in higher node densities.

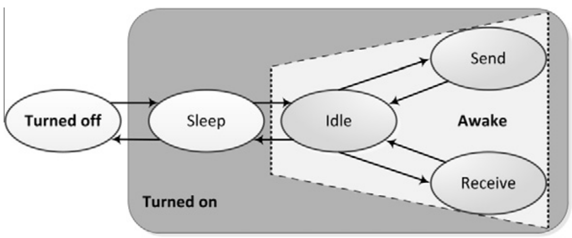


Fig. 1. Different modes of a sensor node in WSNs.

The rest of the paper is structured as follows: Related works are reviewed in Section 2. Theoretical foundations are discussed in Section 3. The proposed MAC protocol is given in Section 4. Section 5 expresses simulation results and finally, Section 6 concludes the paper.

2. Related works

Chao and Lee [8] propose QMAC as a single channel MAC protocol for WSNs utilizing grid quorum to save energy. This protocol tries to prolong network lifetime by increasing node sleep time. However, using only a single channel in a network results in an increase in collision, therefore needing packet retransmission, which increases network energy consumption as well as latency. In addition, although QMAC proposes a method to assign different grid sizes to coronas in constant traffic rate, it is not obvious when and how grid sizes should be changed with traffic rate variations.

PW-MAC [41] is a receiver-initiated predictive wake-up MAC protocol in which every node computes its wake-up times using a pseudo-random wake-up schedule. Each node in PW-MAC periodically wakes up and broadcasts a beacon to announce that it is awake and ready to receive data. A sender has to know a receiver's pseudo-random generator parameters to wake up a little earlier than the receiver does and waits for a beacon. However, PW-MAC has some shortcomings so that each node has to send a beacon every time it wakes up regardless of whether any sender has data to send or not. In addition, protocol overhead increases as each node broadcasts its pseudo-random generator parameters periodically, which in turn worsens at higher network densities.

TMCP [28] is a tree-based multi-channel protocol for data collection applications. The main idea of TMCP is to partition the whole network into multiple vertex-disjoint sub-trees all rooted at a sink. Then, it allocates different channels to each sub-tree and forwards each flow along only its corresponding sub-tree. When a node wants to send information to the sink, it just uploads packets to the sub-tree that it belongs to. TMCP has some shortcomings. It is designed to support data collection traffic and it is difficult to have broadcasts due to its partitions. Aggregation cannot be employed since communication among nodes in different sub-trees is blocked.

EM-MAC [42] is a receiver-initiated multi-channel asynchronous MAC protocol proposed for WSNs. In EM-MAC, a sender rendezvous with a receiver by predicting the wake-up channel and wake-up time of the receiver. In EM-MAC, like PW-MAC, the sender knows the state of the receiver's pseudo-random function used to generate its wake-up channels and times. EM-MAC not only has the shortcomings of PW-MAC but also each node in EM-MAC has to invoke its pseudo-random generator twice, which bears further overhead for the protocol.

The IEEE 802.15.4 [43] protocol, which was originally designed for low-rate Wireless Personal Area Networks (WPANs), can be used for WSN applications. The protocol makes use of multi-channel communication to reduce the effects of interference with co-existing networks. The protocol has two modes of operation: beacon-enabled and beaconless. In the beacon-enabled mode, a coordinator node is responsible for adjusting the channel on which its end-devices should communicate. In this mode, communication can take place in a slotted mode of operation and nodes should directly communicate with the coordinator to get the slot allocations. Even if a node intends to communicate with a peer in its communication range, all transmissions should flow through the coordinator. When the protocol operates in a beaconless mode, it uses CSMA/CA, and nodes function on a fixed channel. Due to the hierarchy in IEEE 802.15.4 networks, the WPAN coordinator is responsible for binding of new nodes, scheduling, and routing in the network. Moreover, since all the nodes in a WPAN communicate on the same channel, in IEEE 802.15.4 contention within the network is not resolved.

CMAC [30] is an asynchronous multi-channel MAC protocol that uses two radios, a Low power wake-up Radio (LR), and a Main half-duplex Radio (MR). LR is always on and it is used to monitor a node's default channel while MR is placed in sleep mode. LR plays two roles: (1) when a node wishes to transmit, the receiver is awakened through a series of pulses, (2) channel negotiation is undertaken before MR is switched on. MR transmits at a constant power level in a predetermined channel, although it can be switched off. CMAC does not require any synchronization, although it needs two transceivers for each sensor node. This increases the hardware complexity and cost of the whole network. Meanwhile, the control channel might also become a bottleneck when many nodes initialize channel negotiation and request data transmission, simultaneously.

MC-LMAC [25] is a synchronous single-radio multi-channel MAC protocol for WSNs. MC-LMAC has been designed with the objective of maximizing the throughput by coordinating transmissions over multiple channels. It is based on the single-channel LMAC [44] protocol. MC-LMAC is schedule-based protocol where nodes switch their interfaces between channels dynamically. Time is slotted and the control over a time slot is assigned to each node to transmit on a particular channel. In fact, a node selects a time slot and a channel on which it is allowed to transmit. The main problem of MC-LMAC is the overhead of its control messages, and it worsens as network density increases.

MMSN is the first multi-frequency MAC protocol designed especially for WSNs [24]. It is based on slotted CSMA where at the beginning of each time slot, nodes need to contend for the medium before they can transmit. MMSN allows users to choose one of the four available frequency assignment strategies such as exclusive frequency, implicit-consensus, eavesdropping, and even-selection [24]. A time slot in MMSN consists of a broadcast contention period and a transmission period. During the contention period, nodes compete for the same broadcast frequency and during the transmission period, nodes compete for shared unicast frequencies. MMSN has some disadvantages. When a node wants to send a data unit, it has to switch between self-frequency and destination frequency at preamble sending time, which increases the message delay and protocol overhead. MMSN has a fixed allocated back-off time in each time slot that is a shortcoming of this protocol. Although MMSN needs time synchronization during media access to provide broadcast support, it does not take full advantages of the synchronization service to resolve the conflicts and/or improve its scheme. There are other multi-channel MAC protocols proposed for WSNs such as Rainbow [23], HyMAC [27], and YMAC [20], most of which do not consider the energy consumption of nodes.

This paper presents Queen-MAC, a quorum-based energy-efficient MAC protocol that independently and adaptively adjusts nodes wake-up times, which reduces the protocol overhead and prolongs network lifetime. Queen-MAC utilizes multiple channels for data transmissions that reduce collisions while providing concurrent transmissions in a broadcast domain. It may be emphasized that the power saving can be achieved at different layers in WSNs, while this paper focuses on the MAC layer solution.

3. Theoretical foundations

This section postulates some assumptions, and then introduces dygrid. The next section presents the details of Queen-MAC. Table 1 lists the general notations used in the paper.

3.1. Assumptions

This work outlines the assumptions made in Queen-MAC protocol development, as follows:

1. Time is divided into a series of time slots.
2. All sensor nodes are homogenous (have a single radio) and have the same transmission range (hop distance) d.
3. Nodes are static in the network.

Table 1
General notations.

Symbol	Description
U	Universal set
n	The cycle length
$V(c, k)$	A v-clique (c, k)
$H(r, k)$	An h-clique (r, k)
h	Shift index
AR	Active ratio
d	Transmission range of a sensor node
E_r	Remaining energy of a sensor node
E_i	Initial energy of a sensor node
T_{MCS}	Mini control slot period
$T_{back\text{-}off}$	Back-off time
g	The number of groups in a network
f	The list of available frequencies
λ	Scale parameter
Γ	Channel rate
P	Packet size
t	Time slot duration

4. Sensor nodes are time synchronized, as assumed in [8,24]. However, nodes can be synchronized locally [45,46] so that each node only needs to be synchronized with its PFs (PF is defined in Definition 4.5). In order to overcome the clock skew, the time synchronization protocol, RTAS [47], can be run periodically to maintain synchronization. RTAS can be used for an ultra low duty-cycle system, and is reported to maintain a 225 μs error bound in an ultra low duty-cycle system [24].
5. Network mission is data collection. All sensor nodes send their data toward the sink in a multi-hop fashion.
6. The direction of broadcast packets is only from the sink.
7. Sensor nodes are uniformly distributed in the sensing environment where the sink is placed in a corner and all sensor nodes are grouped based on their distances (by hop counts) from the sink, as depicted in Fig. 2. To create the node groups, a control packet $hop-notify$ that contains a field $hop-id=0$ is sent from the sink during the network initialization. Upon receiving this control packet, each sensor node increases the $hop-id$ field by one and then resends the packet. A node belongs to group G_i if it receives a $hop-notify$ with $hop-id=i$. If multiple $hop-notify$ packets are received; only the one with the lowest $hop-id$ value is handled, and the rest are discarded. Sensor nodes in G_i are $i+1$ hops away from the sink node and rely on nodes in G_{i-1} to relay their sensed data.

3.2. Dyadic grid quorum system

Quorum systems have been widely used in distributed systems to deal with the mutual exclusion problem [48], fault tolerance, voting [49], and also have been utilized to design protocols for wireless networks [7,8,32–36]. There are several kinds of quorums, such as tree-based [50], majority-based [51], grid-based [37], torus [38], extended torus (e-torus) [32,35]. Some of them, such as grid and torus, have fixed duty cycles that makes them inappropriate for use in a network with different traffic conditions. In addition, the rest (e.g., e-torus that has an adaptive duty cycle) provide a high minimum duty cycle that leads to more energy consumption if used in a network with a low traffic load.

This section proposes a dyadic grid quorum system that provides an adaptive and low duty cycle for sensor nodes, which is highly appropriate for data collection applications.

Consider the sets in which each element denotes a number of a time slot. The following definition expresses a quorum system, in this paper called clique, to propose our quorum system afterwards:

Definition 3.1 (*Clique*). Given an integer n and a universal set $U=\{0,1,\ldots,n-1\}$. Let C be a set of nonempty subsets of U. We call C a *clique* if and only if for all $Q, Q' \in C$, $Q \cap Q' \neq \phi$.

For example, $C=\{\{0,2\},\{1,2,3\},\{0,3\}\}$ is a clique under $U=\{0, 1, 2, 3\}$. The elements of C (i.e., Q) are named quorums. Sensor nodes adopting the quorums of a clique are able to discover each other at least once every n time intervals. In this paper, the definition of a quorum is generalized to define the dyadic grid quorum system.

Definition 3.2 (*Bi-clique*). Given an integer n and a universal set $U=\{0,1,\ldots,n-1\}$. Let X and Y be two sets of nonempty subsets of U. The pair (X, Y) is called a *bi-clique* if and only if for all $Q \in X$ and $Q' \in Y$, $Q \cap Q' \neq \phi$.

For example, for $X=\{\{0, 1\},\{2, 3\},\{0, 1, 3\}\}$ and $Y=\{\{0, 2\},\{1, 3\}\}$, (X, Y) forms a bi-clique.

Definition 3.3 (h-**Clique**(r, k)). Given an integer n and a universal set $U=\{0, 1,\ldots,n-1\}$. Let k, $1 \leqslant k \leqslant \sqrt{n}$, and r, $0 \leqslant r \leqslant n-1$, be two integers. An *h-clique* of r and k is defined as $H(r, k)$:

$$H(r,k)=\left\{\left(\left\lfloor\frac{\sqrt{n}}{k}i\right\rfloor\sqrt{n}+r+j\right)(\bmod\ n) : i=0,\ldots,k-1,\; j=0,\ldots,\sqrt{n}-1\right\}. \tag{1}$$

For instance, when $n=16$, $H(3, 2)=\{3, 4, 5, 6, 11, 12, 13, 14\}$. If n is shown as a $\sqrt{n}\times\sqrt{n}$ grid, $H(3, 2)$ can be illustrated as in Fig. 3a.

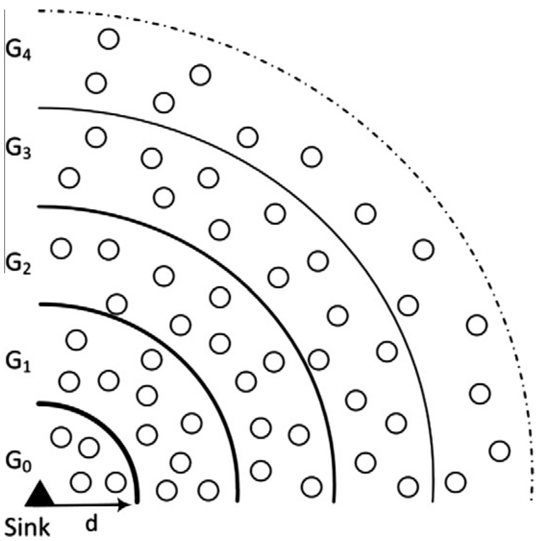


Fig. 2. Sensor nodes are grouped in the network. The ith group is denoted by G_i. The transmission range (hop distance) for all nodes is the same, equal to d.

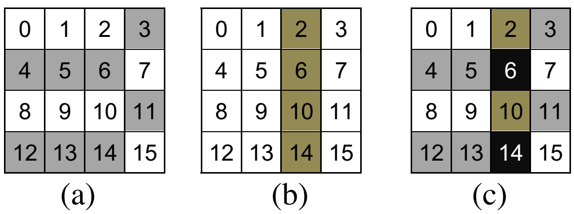


Fig. 3. When $n=16$, (a) an h-clique (3, 2); (b) a v-clique (6, 1); (c) a dygrid (3, 6, 2, 1).

Definition 3.4 (v-**Clique**(c, k)). Given an integer n and a universal set $U = \{0, 1, \ldots, n\text{-}1\}$. Let k, $1 \leqslant k \leqslant \sqrt{n}$, and c, $0 \leqslant c \leqslant n-1$, be two integers. A *v-clique* of c and k is defined as $V(c, k)$:

$$V(c,k) = \left\{ \left(\left\lfloor \frac{\sqrt{n}}{k} i \right\rfloor + c + j\sqrt{n} \right) (\mathrm{mod}\, n) : i = 0, \ldots, k-1, j = 0, \ldots, \sqrt{n}-1 \right\}. \quad (2)$$

For example, when $n = 16$, $V(6, 1) = \{2, 6, 10, 14\}$ as shown in Fig. 3b.

Definition 3.5 (*(r, c, k_1, k_2)-Dyadic grid quorum system*). Given two integers k_1, k_2, $1 \leqslant k_1, k_2 \leqslant \sqrt{n}$, and two arbitrary integers r and c, $0 \leqslant r$, $c \leqslant n-1$. Let X and Y be sets of nonempty subsets of U, where $U = \{0, 1, \ldots, n-1\}$. The pair (X, Y) is called a *(r, c, k_1, k_2)-dyadic grid quorum system* (denoted as *dygrid*(r, c, k_1, k_2)) if and only if (i) X is an h-clique(r, k_1), Y is a v-clique(c, k_2) or vice versa and (ii) the pair (X, Y) is a *bi-clique*.

For example, the *dygrid*(3, 6, 2, 1) for $n = 16$ is shown in Fig. 3c.

Unlike traditional quorum systems, a dygrid quorum system does not guarantee the intersection between h-cliques or v-cliques. However, the intersection between h-cliques and v-cliques are guaranteed. For instance, H (3,2) and V (6,1) that form the *dygrid*(3, 6, 2, 1) have two intersection points 6 and 14 as illustrated in Fig. 3c.

The advantage of the dygrid quorum system is that the size of h-cliques and v-cliques can be considerably smaller. Given the cycle length n, when dygrid is applied to a sensor network, each node may have a duty cycle $O(k\sqrt{n}/n) = O(k/\sqrt{n})$, leading to reduced energy consumption. Fig. 4 compares the minimum duty cycle of dygrid with two commonly used quorum systems (e-torus and grid) in different cycle lengths. As shown in the Fig. 4, dygrid provides a lower minimum duty cycle in different cycle lengths.

Definition 3.6 (*Network Sensibility*). The longest delay for a sensor node to detect another node in its neighborhood is called network sensibility. In a quorum system, it is the distance between two of the farthest intersecting points.

Theorem 3.1. *Given two integers k_1, k_2, $1 \leqslant k_1, k_2 \leqslant \sqrt{n}$, and two arbitrary integers r and c, $0 \leqslant r$, $c \leqslant n-1$. The network sensibility of the dygrid(r, c, k_1, k_2) is*

$$\sqrt{n}(\lceil \sqrt{n}/k_1 \rceil - 1) + \lceil \sqrt{n}/k_2 \rceil.$$

Proof. By Definition 3.3, $H(r,k_1)$ has $k_1 \times \sqrt{n}$ elements where the maximum distance between two successive elements is

$$\sqrt{n}(\lceil \sqrt{n}/k_1 \rceil - 1) + 1. \quad (3)$$

By Definition 3.4, $V(r,k_2)$ has $k_2 \times \sqrt{n}$ elements too, where the maximum distance between two successive elements is

$$\lceil \sqrt{n}/k_2 \rceil. \quad (4)$$

Therefore, the maximum distance of two successive common elements (i.e., network sensibility) of *dygrid*(r, c, k_1, k_2) is the sum of (3) and (4) minus 1 (the common element)

$$\sqrt{n}(\lceil \sqrt{n}/k_1 \rceil - 1) + 1 - \lceil \sqrt{n}/k_2 \rceil - 1 \\ = \sqrt{n}(\lceil \sqrt{n}/k_1 \rceil - 1) + \lceil \sqrt{n}/k_2 \rceil. \quad \square$$

For example, the network sensibility in Fig. 3c is $4(\lceil 4/2 \rceil - 1) + \lceil 4/1 \rceil = 8$. The effect of k_1 and k_2 on the network sensibility is illustrated in Fig. 5.

In a dygrid system, the larger k causes the shorter network sensibility. However, the larger k also leads to a larger duty cycle. Hence, the decision on k should be taken upon node traffic load to save energy. The network sensibilities of three quorum systems at the same duty cycle $\approx 2/\sqrt{n}$ are compared in Fig. 6. By using a dygrid system, network sensibility is less than grid and e-torus systems in different cycle lengths. This means that, sensor nodes using the proposed quorum system can detect their neighboring nodes in less time. Therefore, data can reach the sink with less delay.

The number of rendezvous as another metric is used to compare dygrid with grid and e-torus. For a $\sqrt{n} \times \sqrt{n}$ grid quorum, there are three possible values for the number of rendezvous; at least 2 when each node selects different rows and columns, it is $\sqrt{n}$ when two nodes select the

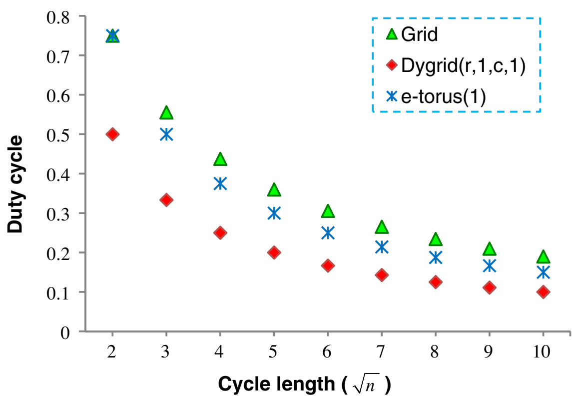


Fig. 4. Minimum duty cycle in different cycle lengths.

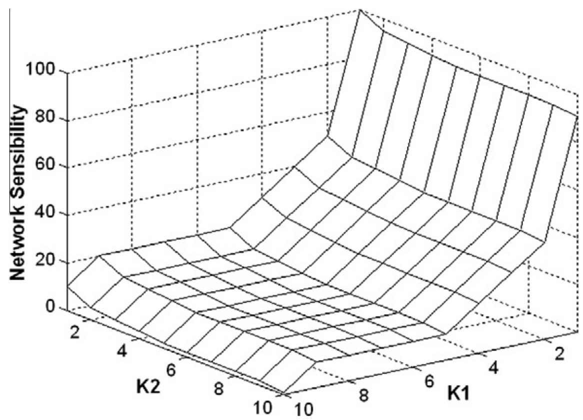


Fig. 5. The effects of k_1 and k_2 on the network sensibility in a dygrid.

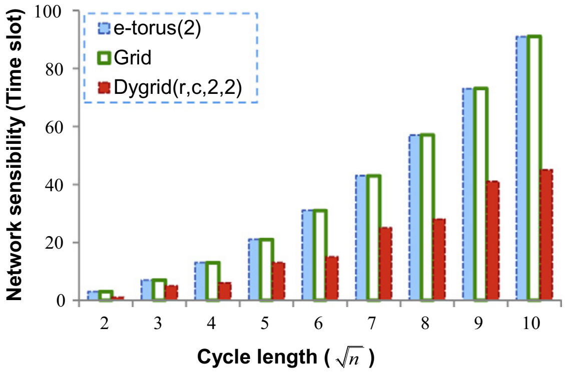


Fig. 6. Network sensibility in different cycle lengths.

same row or column, and at most $2\sqrt{n}-1$ when each node selects the same row and column (i.e., completely overlapped).

For a $\sqrt{n}\times\sqrt{n}$ e-torus quorum system, two quorums e-torus (k_1) and e-torus (k_2) have at least $\lfloor (k_1+k_2)/2 \rfloor$ rendezvous points. The maximum rendezvous occurs when all parameters of two quorums are identical (100% overlap, i.e., $k_1 = k_2$, $r_1 = r_2$, and $c_1 = c_2$) [32]; therefore we have $\sqrt{n}(1+\lfloor k_1/2\rfloor)+(k_1 \mathrm{mod} 2)\lfloor(\sqrt{n}-1)/2\rfloor$.

A $\sqrt{n}\times\sqrt{n}$ dygrid quorum system can present different number of rendezvous depending on k_1 and k_2, that is $\{k_1\times k_2 : 1\leqslant k_1,k_2\leqslant\sqrt{n}, k_1,k_2\in\mathbb{N}\}$. For example, when $n = 16$, there are nine possible values for the number of rendezvous, including 1, 2, 3, 4, 6, 8, 9, 12, 16. Fig. 7 illustrates the effects of k_1 and k_2 on the number of rendezvous in a dygrid.

Hence, it can be concluded that grid has less diversity for the number of rendezvous. It only provides three values. Nevertheless, e-torus and dygrid have a variety of rendezvous points where dygrid offers a better duty cycle at a given rendezvous value. For example, using a dygrid quorum system to provide one rendezvous point, a node will have the duty cycle $1/\sqrt{n}$, whereas for e-torus the duty cycle is $1/\sqrt{n}+\lfloor(\sqrt{n}-1)/2\rfloor/n$ to provide the same number of rendezvous.

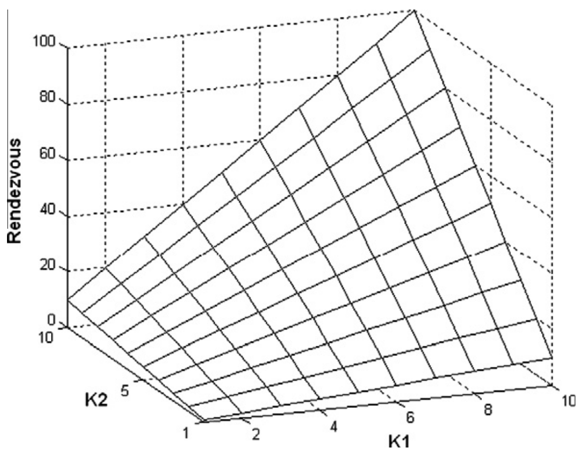


Fig. 7. The effects of k_1 and k_2 on the number of rendezvous in a dygrid.

4. Queen-MAC protocol description

In Queen-MAC, the network topology is considered as Fig. 2, where nodes are grouped based on their distances from the sink. Each node selects *h*- or *v*-clique that depends on its group number. Nodes with an even group number can select *v*-clique and nodes with an odd group number may select *h*-clique or vice versa. By selecting *h*-clique and *v*-clique, two nodes in neighboring groups form a dygrid. In the following, it is proved that it is guaranteed for two nodes *A* and *B*, which select an *h*-clique and a *v*-clique, respectively to form their cycle patterns, meeting each other at least one time during *n* time slots.

Here, two kinds of time slots (quorum and non-quorum time slots) are defined.

Definition 4.1 (*Quorum Time Slot*). The time slot in which a sensor node wakes up to check the medium for a possible data exchange (e.g., time slots 3, 4, 5, 6, 11, 12, 13, and 14 as shown in Fig. 3a).

Definition 4.2 (*Non-quorum Time Slot*). The time slot in which a sensor node can tune its radio into power saving mode (i.e., sleep) to save energy (e.g., time slots 0, 1, 2, 7, 8, 9, 10, and 15 as shown in Fig. 3a).

Definition 4.3 ((*n, m, h*)-**Rotation**). Given positive integers *n*, *m* and *h*, where $h < n$, $n < m$. Let *X* be a subset of the universal set $U = \{0, 1, \ldots, n-1\}$. (n, m, h) – *rotation* of *X* (denoted as $R^n_{m,h}(X)$) is defined as:

$$R^n_{m,h}(X)=\{(x+jn)+h : 0\leqslant(x+jn)+h\leqslant m-1,\ \ \forall x\in X, j\in Z\}. \quad (5)$$

Moreover, all possible rotations of *X* are denoted by $R^n_m(X)=\{R^n_{m,h}(X):\forall h\in U\}$.

A (*n, m, h*)-rotation of *X* is a projection of *X* from the modulo-*n* onto the modulo-*m* plane with an index shift *h*. For example, consider $V(11, 1) = \{3, 7, 11, 15\}$ and $H(8, 1) = \{8, 9, 10, 11\}$, which are subsets of $U = \{0, 1, \ldots, 15\}$. Given two shift indices $h_1 = 3$ and $h_2 = 1$, these two sets can be projected from the modulo-16 plane onto the modulo-31 plane by using $R^{16}_{31,3}(V(11,1))=\{2,6,10,14,18,22,26,30\}$ and $R^{16}_{31,1}(H(8,1))=\{9,10,11,12,25,26,27,28\}$, respectively, as they can be seen in Fig. 8. Note both $R^{16}_{31,3}(V(11,1))$ and $R^{16}_{31,1}(H(9,1))$ are subsets of a new universal set $U' = \{0, 1, \ldots, 30\}$.

Lemma 4.1. *Given integers r, c, k_1, k_2, where* $0\leqslant r,c\leqslant n-1, 1\leqslant k_1,k_2\leqslant\sqrt{n}$, *the pair* $(R^n_m(H(r,k_1)),R^n_m(V(c,k_2)))$ *forms a bi-clique.*

Proof. For brief, let *H* and *V* denote $H(r, k_1)$ and $V(c, k_2)$, respectively. We show that for all *p* and $q, 0\leqslant p,q\leqslant n-1, R^n_{m,p}(H)\cap R^n_{m,q}(V)\neq\phi$. Without the loss of generality, suppose $k_1 = k_2 = 1$. By definition of $V, R^n_{m,q}(V)$ has at least $\sqrt{n}$ elements and any two successive elements must have distance $\sqrt{n}$. Let y_i for $i=0,\ldots,\sqrt{n}$-1, be $\sqrt{n}$ elements of $R^n_{m,q}(V)$, thus, we have

$$q+i\sqrt{n}\leqslant y_i\leqslant(q+i\sqrt{n})+\sqrt{n}-1,\quad i=0\ldots\sqrt{n}-1. \quad (6)$$

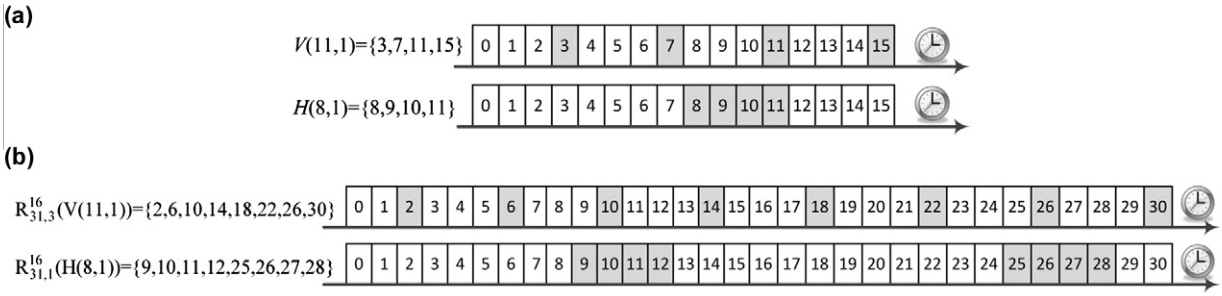


Fig. 8. Overlapping in spite of time slot shifting. (a) Demonstration of $V(11, 1)$ and $H(8, 1)$ for $n = 16$. (b) Demonstration of $R^{16}_{31,3}(V)$ and $R^{16}_{31,1}(H)$ for $m = 31$.

If y_i is included in $R^n_{m,p}(H)$, the proof is finished. Otherwise, it will be proved that at least one y_i must be included in $R^n_{m,p}(H)$. By definition of $H, R^n_{m,p}(H)$ has at least $\sqrt{n}$ elements and any two successive elements in $R^n_{m,p}(H)$ must have distance either 1 or $n-\sqrt{n}+1$. Consider the smallest element x in $R^n_{m,p}(H)$ which is larger than $y' \in R^n_{m,q}(V)$. We have

$$y' + 1 \leqslant x \leqslant y' + n - \sqrt{n} + 2 \tag{7}$$

because any two elements in $R^n_{m,p}(H)$ must have their distance less than or equal to $n-\sqrt{n}+1$. By definition of H, there exists at least $\sqrt{n}$ continuous and ascending elements starting from x in $R^n_{m,p}(H)$. Therefore, by (6) and (7), we have a $y'' \in R^n_{m,q}(V), x \leqslant y'' \leqslant x + \sqrt{n} - 1$, implying that y'' is contained in $R^n_{m,p}(H)$. It goes without saying, for k_1,k_2 > 1, that the same proof can be generalized. □

Definition 4.4. (α-**Cut**). Given a positive integer α and a nonempty set X. $C_\alpha(X)$ is called an *α-cut* of X if and only if $C_\alpha(X) = \{x : 0 \leqslant x \leqslant \alpha - 1, x \in X\}$.

Let Q be a set of nonempty subsets of U. We denote $C_\alpha(Q) = \{C_\alpha(X) : \forall X \in Q\}$.

Theorem 4.1. *Given integers r, c, k_1, k_2, where $0 \leqslant r, c \leqslant n-1, 1 \leqslant k_1, k_2 \leqslant \sqrt{n}$, the pair $(R^n_\infty(H(r,k_1)), R^n_\infty(V(c,k_2)))$ forms a dygrid(r, c, k_1, k_2).*

Proof. By Definitions 4.3 and 4.4, it can be observed that $C_m(R^n_\infty(H(r,k_1))) = R^n_m(H(r,k_1))$ and $C_m(R^n_\infty(V(c,k_2))) = R^n_m(V(c,k_2))$. This theorem is a direct consequence from the Lemma 4.1. □

Suppose two sensor nodes A and B, respectively, adopt $H(8, 1)$ and $V(11, 1)$ as *dygrid*(8, 11, 1, 1) to form their cycle patterns, as shown in Fig. 8a. The above theorem shows that these two nodes are guaranteed to overlap at least one awake time slot within $n = 16$ time intervals, even if there is a time slot shift between sensor nodes, as shown in Fig. 8b.

Theorem 4.2. *Given integers r, c, k_1, k_2, where $0 \leqslant r, c \leqslant n-1, 1 \leqslant k_1, k_2 \leqslant \sqrt{n}$, we have*

$$\left|R^n_n(H(r,k_1)) \cap R^n_n(V(c,k_2))\right| = k_1 \times k_2. \tag{8}$$

Proof. By Definition 3.3, $H(r, 1)$ has a sequence of $\sqrt{n}$ continuous and ascending elements that according to the Lemma 4.1, it has at least one intersection with $V(c,1), \left|R^n_n(H(r,1)) \cap R^n_n(V(c,1))\right| = 1$. Subsequently, $H(r, k_1)$ has k_1 sequences like that, where these are disjoint sequences by definition. Therefore, because each sequence has an intersection with $V(c, 1)$, we have

$$\left|R^n_n(H(r,k_1)) \cap R^n_n(V(c,1))\right| = k_1. \tag{9}$$

Similarly, it can be shown that

$$\left|R^n_n(H(r,1)) \cap R^n_n(V(c,k_2))\right| = k_2. \tag{10}$$

Thus, by (9) and (10) we have

$$\left|R^n_n(H(r,k_1)) \cap R^n_n(V(c,k_2))\right| = k_1 \times k_2. \quad \square$$

As a result, when two neighbor sensor nodes select $H(r,k_1)$ and $V(c,k_2)$ to form a dygrid to schedule their wake-up times, it is guaranteed that these nodes meet each other in $k_1 \times k_2$ time slots per n time slots.

4.1. Wake-up schedule

In Queen-MAC, power saving is attained by reducing the number of awake times. It determines the wake-up frequency for each sensor node based on its own traffic load. Dygrid quorum system is utilized to represent the quorum time slots, in which a sensor node must be awake. Here, it is explained how to decide on k for each sensor node. As shown in Fig. 2, a sensor node from a closer group to the sink, e.g., G_1, has more traffic load than nodes in a farther group, e.g., G_2, because nodes in a closer group have to relay traffic from farther groups, in addition to its own traffic. This is true for all groups except last group, e.g., G_4 in Fig. 2, because it is only responsible for its own traffic transmission. Groups' areas are calculated so that to be able to find the average number of farther nodes for which a closer node is responsible to pass along their data.

According to the network model shown in Fig. 2, the area of G_0 is (1/4) πd^2, and the area of G_1 is (1/4) $\pi(2d)^2$-(1/4)πd^2 = (3/4) πd^2, where d is the transmission range of nodes. The area of the other groups can be calculated in a similar manner. Generally, the area of G_i is $((2i+1)/4)\pi d^2$. The ratio for the area of G_1 to G_0 is $(G_1/G_0) = 3$. In general, the ratio for area of G_{i+1} to G_i is $(G_{i+1}/$

$G_i) = (2i + 3)/(2i + 1)$. This means that, on average, a sensor node in G_i is responsible for relaying traffic for $(2i + 3)/(2i + 1)$ nodes in G_{i+1}.

Now, assume that each sensor node requires to transmit x packets for any report. Therefore, a node in G_3, for example, has to forward $9x/7$ packets from G_4 besides its own data, where it sums up to $16x/7$ packets. Similarly, a sensor node in G_2, in addition to its own data, is required to forward $16x/5$ packets from G_3. Generally speaking, in a WSN with g groups, each sensor node in group i is responsible to forward F_i packets where F_i is

$$F_i = x + \left(\frac{2i+3}{2i+1}\right)F_{i+1}, i = 0, \ldots g-1, F_g = 0. \quad (11)$$

It can be summarized that, the nodes in inner groups (closer to the sink) which have more traffic load, can choose a larger k than sensor nodes in outer groups (farther from the sink) which have a lighter traffic load. The larger k for a sensor node results in more quorum time slots, during which the sensor node is awake and can send/receive data.

Theorem 4.3. *Let sensor nodes channel rate be Γ bps, time slot duration be t seconds, and packet size be P bits. In a constant traffic load x, a node in group i should select its initial k_i as follows:*

$$k_i = \left\lceil \frac{1}{\sqrt{n}} \left(\left\lceil \frac{P \times n \times (F_i - x)}{\Gamma} \right\rceil + \left\lceil \frac{P \times n \times F_i}{\Gamma} \right\rceil \right) \right\rceil \quad (12)$$

Proof. With channel rate Γ bps, time slot duration t seconds, and packet size P bits, a sensor node can send/receive at $\frac{\Gamma \times t}{P}$ packets per time slot. We know from (11) that a sensor node in group i is responsible to forward F_i packet/s; therefore, it will have $(n \times t) \times F_i$ packets to forward in a cycle length n (i.e., receiving $(n \times t)(F_i - x)$ packets and sending $(n \times t) \times F_i$ packets). Thus, a sensor node in group i needs $\left\lceil \frac{P \times n \times (F_i - x)}{\Gamma} \right\rceil + \left\lceil \frac{P \times n \times F_i}{\Gamma} \right\rceil$ time slots per cycle to forward its traffic. Consequently, it should select its k_i as:

$$k_i = \left\lceil \frac{1}{\sqrt{n}} \left(\left\lceil \frac{P \times n \times (F_i - x)}{\Gamma} \right\rceil + \left\lceil \frac{P \times n \times F_i}{\Gamma} \right\rceil \right) \right\rceil \quad \square$$

For example, when $\Gamma = 250$ kbps, $P = 32$ bytes, $x = 10$ packet/s, $g = 5$ groups, and $n = 36$, a node in group $i = 0$ selects

$$k_o = \left\lceil \frac{1}{\sqrt{36}} \left(\left\lceil \frac{32 \times 8 \times 36 \times (250 - 10)}{250 \times 1024} \right\rceil + \left\lceil \frac{32 \times 8 \times 36 \times 250}{250 \times 1024} \right\rceil \right) \right\rceil = 3,$$

and a node in group $i = 2$ selects

$$k_2 = \left\lceil \frac{1}{\sqrt{36}} \left(\left\lceil \frac{32 \times 8 \times 36 \times (42 - 10)}{250 \times 1024} \right\rceil + \left\lceil \frac{32 \times 8 \times 36 \times 42}{250 \times 1024} \right\rceil \right) \right\rceil = 1.$$

Therefore, according to the above theorem, a sensor node in group i starts with k_i. But, when a node's traffic has been changed, that node has to adjust its k. These changes may happen after each cycle and obey the following rules:

- If the remaining packets are more than $(\Gamma \times t)/P$, then k should be increased by one.
- If no packets are left and the number of forwarded packets in that cycle is less than or equal to $\frac{\Gamma \times t \times \sqrt{n}}{2P}(k_i - 1)$, then k should be decreased by one.

However, if no data has been sent in a cycle due to collision, then r or c depending on the node's clique (i.e., h-clique(r, k) or v-clique(c, k)) should be changed randomly.

Definition 4.5 (*Possible Forwarder (PF)*). For a given node A in group G_i, $PF(A)$ is defined as the set of nodes in group G_{i-1} that are in the transmission range of node A.

For instance, as can be seen in Fig. 9, $PF(A) = PF(B) = \{D, E, F\}$ and $PF(C) = \{D, S, V, W\}$.

Definition 4.6 (*Active Ratio (AR)*). The ratio by which a sensor node has to keep its radio in awake mode is called active ratio. It can be measured by the ratio of the number of quorum time slots to the cycle length. Given an integer k for an h-clique (or a v-clique) in a dygrid, the $AR(k)$ is:

$$AR(k) = \frac{k \times \sqrt{n}}{n} = \frac{k}{\sqrt{n}} \quad (13)$$

Theorem 4.4. *Sensor node A in G_i needs to transmit toward the sink. When it wakes up, the probability, p_i^A, of finding at least one node awake in G_{i-1} is*

$$P_i^A = 1 - (1 - AR(k_{i-1}))^{t_a} \quad (14)$$

where $|PF(A)| = t_a$, and the nodes in group G_{i-1} select k_{i-1}.

Proof. In each n time slot, according to (13), the probability that one node of t_a nodes stays in awake mode is $AR(k_{i-1})$. Therefore, the probability that node stays in sleep mode is $1 - AR(k_{i-1})$. The probability that all t_a nodes remain in sleep mode is $(1 - AR(k_{i-1}))^{t_a}$. Hence, p_i^A, the probability of finding at least one of t_a nodes to be awake, is $(1 - (1 - AR(k_{i-1}))^{t_a})$. $\square$

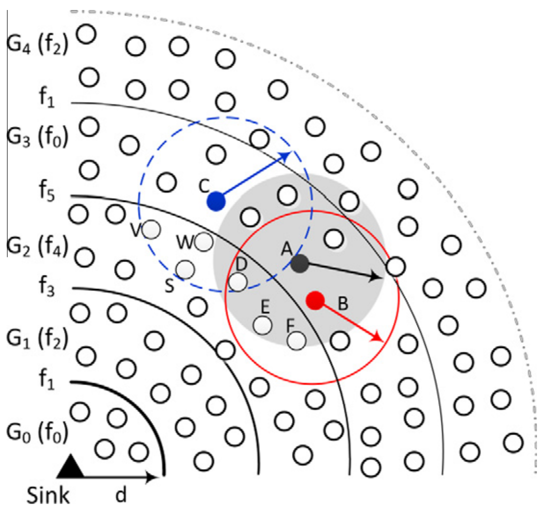


Fig. 9. The PF of nodes, PF (A) = PF (B) = {D, E, F}, PF (C) = {D, S, V, W}.

Theorem 4.4 indicates that a larger k and PF for a node (in a group like G_i) results in a larger probability to meet a node in the immediate closer group (e.g., G_{i-1}). Consequently, it causes a higher probability to transmit toward the sink.

4.2. Channel assignment

Queen-MAC utilizes multiple channels to transmit packets, concurrently. Channel assignment in Queen-MAC is straightforward. It has a low overhead for the protocol. In the initial phase, the sink puts the available frequencies into a list (called f) and broadcast the list to the network (It can put the list into the $hop-notify$ control packet, which is used in Section 3.1). Each group of nodes, except boundary groups, such as G_0 and G_4 in Fig. 9, selects four different frequencies as broadcast and unicast frequencies. Sensor nodes exploit two broadcast frequencies. A Frequency to Receive Broadcast packets (F_{rb}) from the immediate closer group, and a Frequency to Send Broadcast packets (F_{sb}) to its immediate farther group. Moreover, it employs two unicast Frequencies to Send/Receive Unicast packets (F_{su} & F_{ru}) to/from the neighboring groups. Boundary groups select the fewer number of frequencies; the nodes in the farthest group of the network (e.g., G_4 in Fig. 9) do not need to switch to their $F_{ru}(F_{sb})$ to receive unicast (send broadcast) packets from (to) the farther group. In addition, the nodes in the closest group to the sink (i.e., G_0) do not need to switch to F_{su} to send unicast packets to the sink. It can reuse F_{rb} (which is used for receiving broadcast packets) to send unicast packets. In fact, Queen-MAC channel assignment occurs as follows:

(i) F_{rb}: Frequency $f(2i \bmod 6)$ is assigned to the nodes that belong to G_i for receiving broadcast packets from G_{i-1}.
(ii) F_{sb}: Frequency $f((2i+2) \bmod 6)$ is assigned to the nodes that belong to G_i for sending broadcast packets to G_{i+1}.
(iii) F_{ru}: Frequency $f((2i+1) \bmod 6)$ is assigned to the nodes that belong to G_i for receiving unicast packets from the nodes in group G_{i+1}.
(iv) F_{su}: Frequency $f((2i-1) \bmod 6)$ is assigned to the nodes that belong to G_i for sending unicast packets to the nodes in group G_{i-1}.

This channel assignment ensures 2-hop frequency separation for groups of nodes. The network can operate with six channels, as shown in Fig. 9. This simple channel assignment method has very low overhead and simply depends on the group number of nodes (i.e., i in G_i).

As illustrated in Fig. 10, unicast packet transmissions to neighboring groups are done without colliding with each other. When a node sends unicast packets, it is free from collision with ongoing broadcasts in the neighboring groups that take place on broadcast frequencies. Moreover, broadcasting in different groups is also free from collision with each other.

There are two situations in Queen-MAC in which collision may happen in unicast packet transmissions. First, when two nodes from the same group that have the same PF (such as nodes A and B shown in Fig. 9) send data simultaneously. After a certain number of time slots, if collision still exists, two nodes can keep themselves away from collision by reselecting r or c (depending on selected clique) randomly.

The next possible collision situation occurs when two nodes (with different PFs) send data simultaneously, where the intersection of their PFs is nonempty (such as nodes A and C shown in Fig. 9, $PF(A) \cap PF(C) = \{D, E, F\} \cap \{D, S, V, W\} = \{D\}$). In such a case, simultaneous transmissions may cause collisions only at common nodes (here, node D in Fig. 9), but non-common nodes (such as nodes S, V, and W for node C, and nodes E and F for node A in Fig. 9) can receive data if they are awake.

Theorem 4.5. *Let A be a node in group G_i where $|PF(A)| = t_a$, and each node in G_j selects k_j. Furthermore, let the possibility be the same for a node being on the assigned frequencies. When A wakes up to transmit toward the sink, the probability of collision occurring for A, p_c^A, and no chance to transmit is*

$$P_c^A = \frac{(1-(1-AR(k_i))^{w_a})(1-(1-AR(k_{i-1}))^{t_a})}{\psi_i \times \psi_{i-1}} \quad (15)$$

where w_a is the number of neighbors of node A with PFs equal to $PF(A)$, ψ_j is the number of frequencies used in G_j.

Proof. According to (14), the probability of finding at least one of t_a node in G_{i-1} awake and in the frequency F_{ru} is $(1-(1-AR(k_{i-1}))^{t_a})/\psi_{i-1}$. Similarly, the probability that at least one of w_a is awake and is tuned to the same unicast frequency (i.e., F_{su}) is $(1-(1-AR(k_i))^{w_a})/\psi_i$. Hence, p_c^A, the probability of occurring collision is

$(1-(1-AR(k_i))^{w_a})(1-(1-AR(k_{i-1}))^{t_a})/(\psi_i \times \psi_{i-1})$. □

4.3. Data communication

The data transmission of Queen-MAC follows the four-way RTS/CTS/DATA/ACK dialog, where the ACK message is

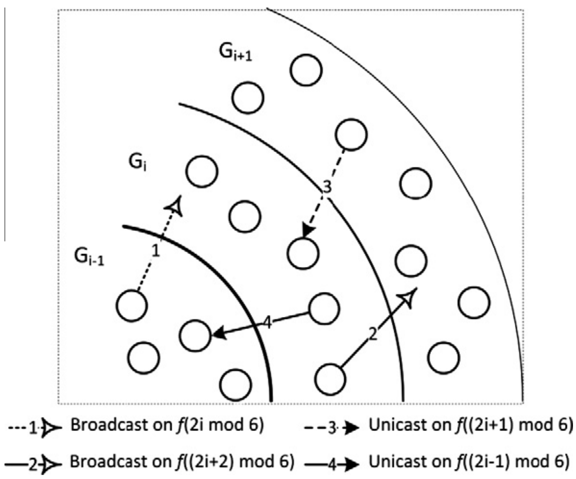


Fig. 10. Possible concurrent collision-free transmissions in a collision domain.

sent on demand. Each node can only wake up and transmit in its quorum time slots. Nodes should sleep in their non-quorum time slots so they can buffer every sensed data. Here, two types of packets are considered: broadcast and unicast packets. Broadcast packets have higher priority than unicast packets in Queen-MAC. Hence, each node first checks to receive/send possible broadcast packets when it wakes up. It can receive broadcast packets (at assigned F_{rb} frequency) from the nodes in its immediate group, closer to the sink. It also can transmit broadcast packets to nodes in its immediate farther group, at assigned F_{sb} frequency. For unicast packets, each node in group G_i, except the farthest group, can receive packets from nodes in group G_{i+1}, and then forward those packets to nodes in group G_{i-1}, where the receiver for nodes in group G_0 is the sink. The structure of a quorum time slot in Queen-MAC is shown in Fig. 11.

Each time slot in Queen-MAC consists of two parts, Mini Control Slots (MCS) and Data. The number of MCSs for a network with g groups can be at most $g + 2$. Each node in a group selects three MCSs for checking to receive possible broadcast, send data (unicast or broadcast, if any), and receive potential unicast data, respectively. For example, node A in group G_i selects three successive mini control slots $i - 1$, i, and $i + 1$ to probe possible data transmission. Next, the detailed data communication process (for a node when it wakes up in its quorum time slot) is given:

1. When a quorum time slot for a node in group G_i is received, it wakes up on MCS $i - 1$ to snoop on its F_{rb} frequency (i.e., $f(2i \bmod 6)$). If the channel is busy, it becomes aware that another node is broadcasting a packet. Therefore, it should receive the broadcast packet(s) during the rest of the quorum time slot, as shown in Fig. 12a. Otherwise, it goes to the next MCS.
2. On the next mini control slot (i.e., MCS i), if the node has broadcast data to send in group G_{i+1}, it switches to its F_{sb} frequency (i.e., $f((2i + 2) \bmod 6)$) and transmits the broadcast data, as depicted in Fig. 12b. Otherwise, if the node has unicast data to send toward the sink, it switches to its F_{su} frequency (i.e., $f((2i - 1) \bmod 6)$) and transmits a RTS for selecting one node in its *PF* to transfer the data. In the selection process, all the awake nodes in group G_{i-1} that have received the RTS should back off before sending CTS. The back-off time is based on each node's residual energy, such that

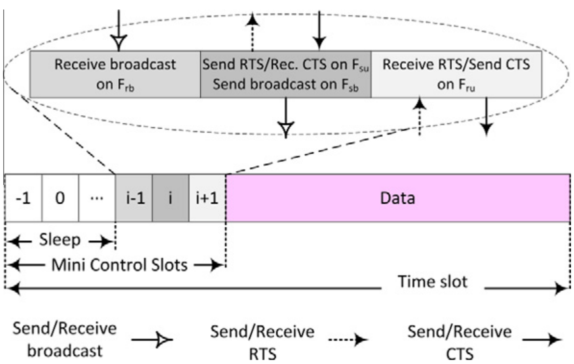


Fig. 11. The structure of a quorum time slot for a node in G_i.

$$T_{back-off} = \lambda\left(1 - \frac{E_r}{E_i}\right)T_{MCS} \tag{16}$$

where E_r is the residual energy, E_i is the initial energy of a sensor node, T_{MCS} is equal to a MCS time period and λ, $0 < \lambda < 1$, is the scale parameter. That is to say, the node with higher residual energy is the candidate to receive the unicast data. After selecting a suitable node in group G_{i-1} (upon receiving the first CTS), data are sent to it.

3. If there are no data to send, or no CTS has been received, the node goes to the next control slot (i.e., MCS $i + 1$) to snoop on the frequency $f((2i + 1) \bmod 6)$ to receive possible unicast data from the group G_{i+1}. The node waits for a RTS during the MCS period. If the node receives a RTS, it backs off, then responds with CTS and waits to receive data. If it does not receive any RTS, or if it receives the first packet with the receiver address different from its address (i.e., it has not been selected for data transmission by the sender), it goes into sleep mode until the next quorum time slot. Otherwise, if the first received unicast packet has the same receiver address as the node address; it receives unicast data in the rest of the time slot, as illustrated in Fig. 12c.
4. When a node in group G_i wakes up, and has no data either to send or to receive, it goes to sleep mode until the next quorum time slot, as shown in Fig. 12d.

5. Performance evaluation

To evaluate the performance of Queen-MAC, the simulator OPNET Modeler 14.0 [40] is used. Queen-MAC is compared with two other protocols, QMAC [8] and TMCP [28]. QMAC is a grid quorum-based MAC protocol for WSNs that uses only one channel for data transmission toward the sink. TMCP is a tree-based multi-channel MAC protocol for data collection in WSNs. It partitions the network into multiple vertex-disjoint sub-trees all rooted at the sink and allocates different static frequencies to each sub-tree and forwards each flow only along its corresponding sub-tree.

Because Queen-MAC utilizes six channels, which are noise-free, we use six channels to compare it fairly with

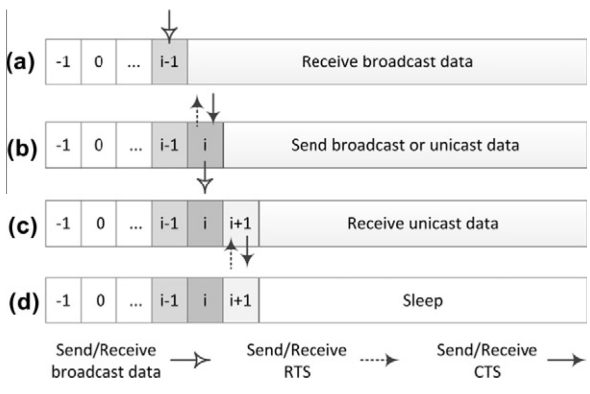


Fig. 12. Different communication types and frequency switching for a node in group G_i. (a) When the node receives broadcast data. (b) When the node has unicast/broadcast data to send. (c) When the node receives unicast data. (d) When the node has no data to send/receive.

TMCP. We also compare QMAC with Queen-MAC to show the efficiency of the proposed quorum system where QMAC uses grid quorum and Queen-MAC utilizes our proposed dygrid quorum system. QMAC does not support more than one channel.

The characteristics of CC2420 [52] are utilized to simulate the radio of sensor nodes, so that its available frequencies start at the initial frequency 2405 and end at 2480 MHz. The bandwidth of each channel is 5 MHz. Central frequency of each channel is calculated as follows:

$$f_i = (2405 + 5 \times i)\text{MHz}, \quad i = 0, 1, 2, \ldots, 15. \tag{17}$$

The simulation is conducted in a network with five groups of nodes where sensor nodes are uniformly placed within an area of radius 350 m. The transmission range of a sensor node is 75 m unless otherwise mentioned. The channel capacity is 250 Kbps. Each node generates a 32-byte data packet every second. A time slot is set to be 100 ms long. The energy consumption model of MICAz [53] is employed in the simulation, where the power consumption for transmit, receive, idle, and sleep modes are 52.2 mW, 83.1 mW, 105 µW, and 48 µW, respectively. Each sensor node has an initial energy of 10 J. Each simulation run lasts 1000 s (10,000 time slots). A spot in the subsequent figures shows an average of 10 simulation runs. For each data value in the results, its 90% confidence interval has also been given. Table 2 summarizes the default simulation parameters.

Observations have been generated from the following aspects: energy consumption, transmission latency, transmission success ratio, the impact of node density, traffic load, and number of groups on the mentioned network parameters.

Table 2
Default simulation parameters.

Parameter	Value	Parameter	Value
Number of groups (g)	5	Initial energy (E_i)	10 J
Transmission range (d)	75 m	Transmit power	52.2 mW
Packet size (P)	32 bytes	Receive power	83.1 mW
RTS packet size	2 bytes	Idle power	105 µW
CTS packet size	3 bytes	Sleep power	48 µW
ACK packet size	3 bytes	Channel rate (Γ)	250 kbps
Cycle length (n)	36	Time slot size (t)	100 ms
Node number (N)	120	T_{MCS}	1 ms
λ	0.7	Application	CBR streams
Node placement	Uniform	Source rate (x)	1 packet/s

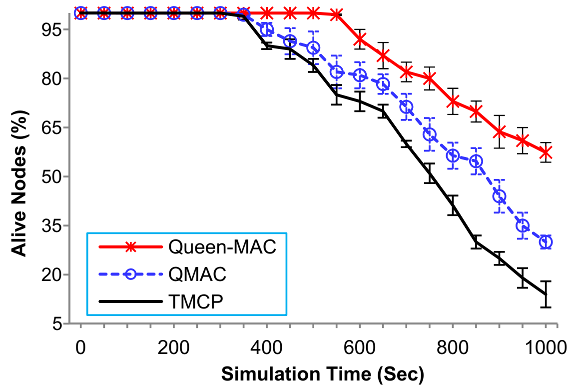


Fig. 13. The effect of different protocols on the percentage of alive nodes.

5.1. Impact on energy consumption

The efficiency of the protocols to increase network lifetime is compared with each other. Fig. 13 illustrates that Queen-MAC has increased the network lifetime against QMAC and TMCP. The reason is that QMAC protocol only uses a single frequency for sending data, so more collisions occur and packet retransmissions lead to more energy consumption. In TMCP, because each node can only send data to its single parent node, the time that it has to wait for its parent to get ready to receive, causes more energy consumption. In addition, due to the tree topology of TMCP, the children of a node have to send data only to its single parent that leads to early energy depletion of the parent node.

Fig. 14 illustrates the average energy consumption for different increasing traffic loads where simulations run for 1000 s. It shows that the average energy consumption is increased in the three mentioned protocols. However, Queen-MAC results in the lowest energy consumption for different traffic loads. Queen-MAC has lower average energy consumption than TMCP owing to the usage of dygrid for adaptive wake-up scheduling. It surpasses QMAC due to utilization of multiple channels for data communication and benefits from dygrid.

5.2. Impact on transmission latency

The latency metric has been applied for comparison of three protocols, Queen-MAC, TMCP, and QMAC, to measure their relevant latency, as depicted in Fig. 15. It illustrates that overall latency for wireless traffic is lower for Queen-MAC than TMCP and QMAC. The reason is that Queen-MAC utilizes multiple frequencies to forward frames toward the sink. Therefore, frames are received at

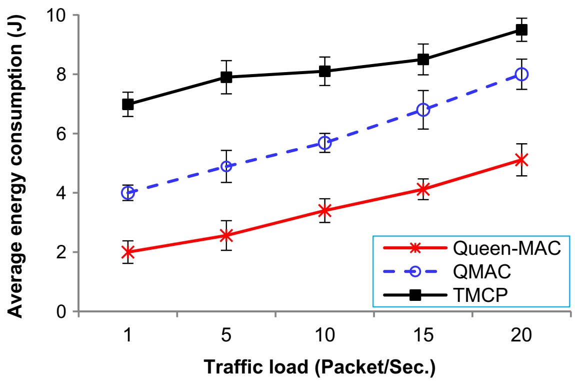


Fig. 14. The effect of different protocols on average energy consumption under various traffic loads.

the sink with lower latency than QMAC. QMAC uses a single frequency that causes more collisions and needs retransmission, which increases delay. The other reason that Queen-MAC surpasses QMAC in traffic latency is the utilization of the dygrid quorum system. It has a better adaptability with network traffic changes than the grid quorum system (used in QMAC). In Queen-MAC, each node has the chance to forward its data to more than one node (called PFs), which results in lower delay for Queen-MAC than TMCP, which has only one possible node to forward its data. However, as it can be seen in the Fig. 15, TMCP has better latency than the other two protocols at the time interval 0–400 s. This is because, in TMCP, each node wakes up every time it has data to transmit. It waits for its parent to get ready to receive the data.

Therefore, the data are received to the sink with less latency at 0–400 period in the simulations. Nevertheless, because of the bottleneck in the parent nodes and the idle listening problem in children nodes, they waste energy and finally die out early, which lead to more latency in the network from time 400 s in the simulations.

Fig. 16 illustrates the average latency for different increasing traffic loads in the sensor network. It shows that the average delay is increased in the three protocols. Nevertheless, Queen-MAC results in the lowest average latency for different traffic loads due to utilizing dygrid.

5.3. Impact on the transmission success ratio

Delivery ratios of the protocols are compared with each other in Fig. 17. The delivery ratio is defined as the ratio of the amount of packets being correctly received at the sink to the amount being sent by all the senders. As it can be seen in Fig. 17, at the first period of time (0–200 s), TMCP has a better delivery ratio than two other protocols due to utilization of multiple channels to forward data. As simulations are continued, TMCP delivery ratio is decreased because of its higher energy consumption, which causes early energy depletion of nodes. However, Queen-MAC has a better overall delivery ratio than the other two protocols.

In Fig. 18, the effect of different protocols is illustrated under various traffic loads. Simulations run for 1000 s where each node has initial energy 10 J. The average delivery ratio is calculated for 600 s. Queen-MAC has a better average delivery ratio than QMAC or TMCP under different traffic loads.

5.4. Effect on the number of groups

To compare the effect of different number of groups on delivery ratio and latency of Queen-MAC under various traffic loads, simulations for 4, 5, and 6 groups of nodes are run in various traffic loads from 1 packet/s. to 20 packet/s. with an interval of 5 for each (except 1). As it can be seen in Fig. 19, delivery ratio is decreased with increasing the number of groups (in different traffic loads). When the number of groups is increased, the nodes in groups closer to the sink have to forward more traffic toward the sink. This leads to more collisions that decrease the delivery ratio in the network.

Fig. 20 illustrates the latency of Queen-MAC under various traffic loads in different groups of nodes. The average latency is increased with the increase of the number of groups. As mentioned before, when the number of groups is increased, collisions in the closer groups are increased. In addition, the traffic of farther groups has to traverse more hops to reach the sink that leads to further delay.

5.5. Impact of node density

In this section, we evaluate Queen-MAC's performance when different node densities are utilized. The source rate is considered as 1 packet per second. The node density is increased from 22 to 34, by adjusting different radio ranges. The node density of a network with N nodes can be calculated as follows:

$$Density(R) = \frac{N\pi d^2}{A} \tag{18}$$

where d is nodes' radio transmission range, and A is the terrain area.

It is observed that the average delivery ratio in QMAC, TMCP, and Queen-MAC is decreased with the more nodes within two hops, as shown in Fig. 21. When more nodes participate in a communication, congestion is high; hence,

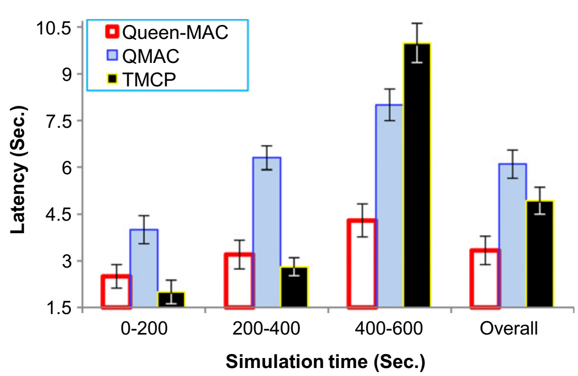


Fig. 15. The effect of different protocols on latency (source rate = 1 - packet/s.).

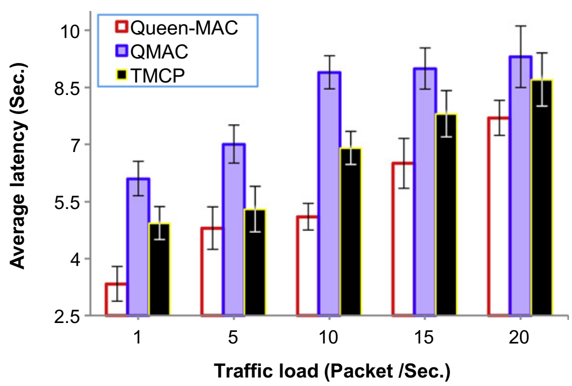


Fig. 16. The effect of different protocols on latency under various traffic loads.

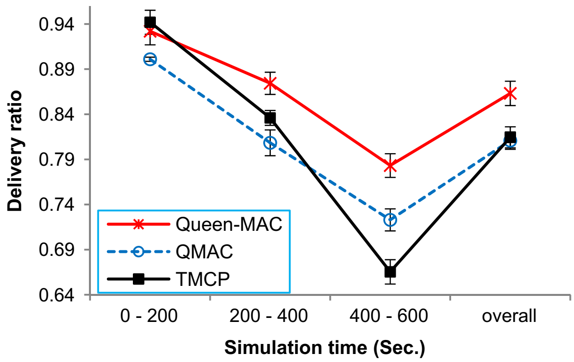


Fig. 17. The effect of different protocols on delivery ratio (source rate = 1 packet/s.).

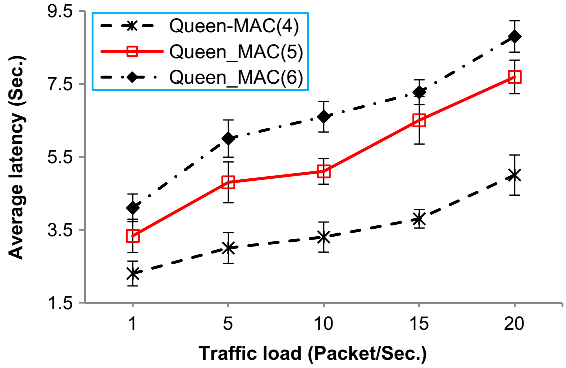


Fig. 20. The effect of Queen-MAC on latency in the networks with different groups of nodes.

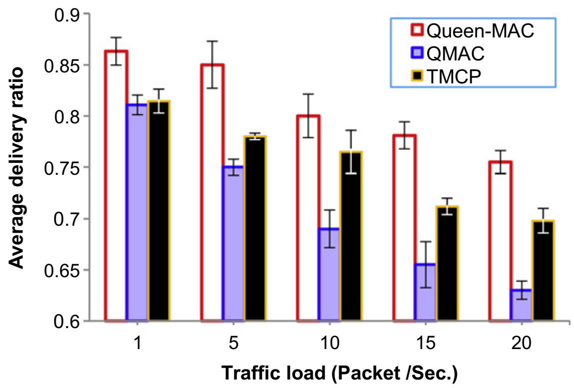


Fig. 18. The effect of different protocols on delivery ratio under various traffic loads.

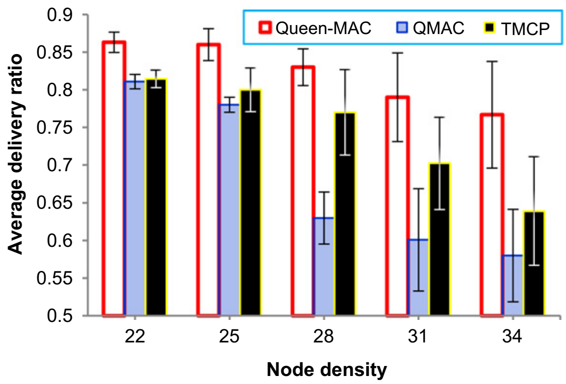


Fig. 21. The effect of different protocols on delivery ratio in the networks with different node densities.

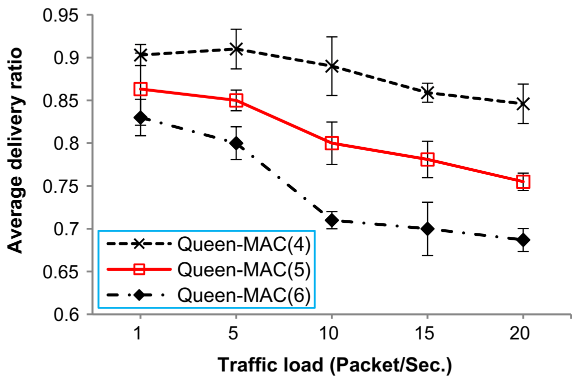


Fig. 19. The effect of Queen-MAC on delivery ratio in the networks with different groups of nodes.

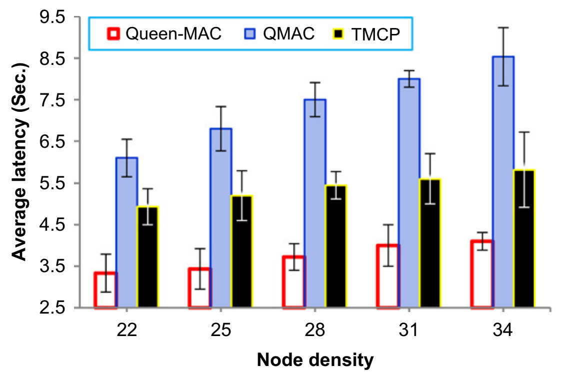


Fig. 22. The effect of different protocols on latency in the networks with different node densities.

packet loss and latency are increased. By utilizing multiple channels, Queen-MAC has a better delivery ratio than QMAC that uses only a single channel. In addition, Queen-MAC can change its dygrid parameters (i.e., r and c) to change wake-up times of nodes trying to keep the nodes away from collision. Queen-MAC also provides better delivery ratio than TMCP in various node densities. This is because of the tree structure of nodes in TMCP that leads to early energy depletion of parent nodes and it makes the average delivery ratio lower than Queen-MAC.

As illustrated in Fig. 22, the average latency of different protocols is increased with increasing the node density of network. Queen-MAC surpasses the other two protocols. As mentioned earlier in Section 5.2, the reasons are the low network sensibility and the adaptability of Queen-MAC to network traffic changes by adjusting the dygrid

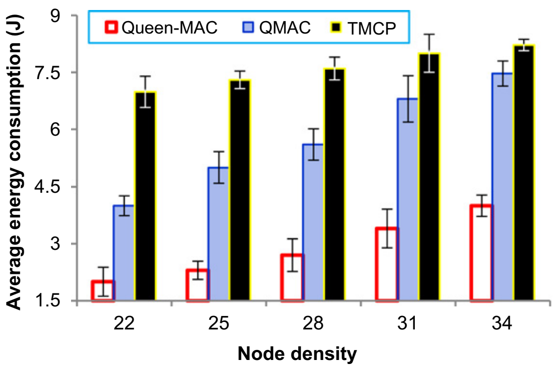


Fig. 23. The effect of different protocols on energy consumption in the networks with different node densities.

parameter *k*, utilizing multiple channels, and selecting a node with more energy from PFs for data transmission.

The average energy consumption of the protocols in different node densities is compared with each other in Fig. 23. Although average energy consumption is increased with increasing density of nodes, Queen-MAC due to the proposed dygrid (which is an adaptive and low duty cycle quorum) has better energy consumption than TMCP and QMAC.

6. Conclusions and future works

In this paper, a new quorum system, named dygrid, is proposed that provides an adaptive and low duty cycle. Dygrid can provide various rendezvous points and decrease network sensibility when used in a network. A network model for data collection applications is proposed and analyzed. Moreover, a lightweight channel assignment method is also proposed, which only depends on the group number of nodes. Dygrid quorum system and the proposed channel assignment method are utilized to propose an adaptive energy-efficient MAC protocol called Queen-MAC for data collection applications in wireless sensor networks, even though it can be used for data dissemination as well. Queen-MAC provides multi-channel access to the medium while it manages sleep/wake-up of nodes, independently. Theoretical analyses are given to evaluate the performance of Queen-MAC, which demonstrate that the proposed protocol is more energy efficient while providing better network latency. Finally, simulations in OPNET Modeler 14.0 show that Queen-MAC increases the network lifetime while it reduces network latency. Simulations indicate that Queen-MAC surpasses QMAC and TMCP in various traffic loads while provides a lower average latency and higher delivery ratio. It also shows that the performance of Queen-MAC is more significant in higher node densities.

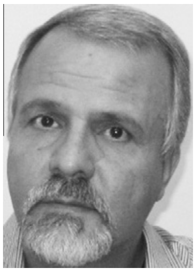
With the proposed channel assignment method that has low overhead for the protocol, Queen-MAC can work in an environment where six channels from the sixteen channels are available. As a future project, we are working on a more flexible method for channel assignment that may have higher overhead, but it can utilize more channels. In such channel assignment, nodes utilize the proposed channel assignment method to assign control channels. Then, nodes negotiate on these channels to choose the available free channels for data communication. In addition, utilizing quorum systems for channel assignment is another issue for which we are planning research.

References

[1] I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless sensor networks: a survey, Computer Networks 38 (2002) 393–422.
[2] C. Bong Jun, S. Xuemin, Adaptive asynchronous sleep scheduling protocols for delay tolerant networks, IEEE Transactions on Mobile Computing 10 (2011) 1283–1296.
[3] S.H. Wu, C.M. Chen, M.S. Chen, Collaborative wakeup in clustered ad hoc networks, IEEE Journal on Selected Areas in Communications 29 (2011) 1585–1594.
[4] Z.T. Chou, Y.H. Lin, R.H. Jan, Optimal asymmetric and maximized adaptive power management protocols for clustered ad hoc wireless networks, IEEE Transactions on Parallel and Distributed Systems 22 (2011) 1961–1968.
[5] W. Shan-Hung, C. Chung-Min, C. Ming-Syan, An asymmetric and asynchronous energy conservation protocol for vehicular networks, IEEE Transactions on Mobile Computing 9 (2010) 98–111.
[6] S. Lai, B. Ravindran, H. Cho, Heterogenous quorum-based wake-up scheduling in wireless sensor networks, IEEE Transactions on Computers 59 (2010) 1562–1575.
[7] Y.C. Kuo, Quorum-based power-saving multicast protocols in the asynchronous ad hoc network, Computer Networks 54 (2010) 1911–1922.
[8] C.M. Chao, Y.W. Lee, A quorum-based energy-saving MAC protocol design for wireless sensor networks, IEEE Transactions on Vehicular Technology 59 (2010) 813–822.
[9] Y. Wei, J. Heidemann, D. Estrin, An energy-efficient MAC protocol for wireless sensor networks, in: INFOCOM 2002. Proceedings of Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 1563, IEEE, 2002, pp. 1567–1576.
[10] Y. Dongyu, Q. Ying, L. Shining, L. Zhigang, RW-MAC: An asynchronous receiver-initiated ultra low power MAC protocol for wireless sensor networks, in: IET International Conference on Wireless Sensor Network, 2010. IET-WSN, 2010, pp. 393–398.
[11] M. Buettner, G.V. Yee, E. Anderson, R. Han, X-mac: a short preamble mac protocol for duty-cycled wireless sensor networks, in: 4th International Conference on Embedded Network SenSys, Boulder, CO, October 2006, pp. 307–320.
[12] S. Ray, I. Demirkol, W. Heinzelman, ADV-MAC: analysis and optimization of energy efficiency through data advertisements for wireless sensor networks, Ad Hoc Networks 9 (2011) 876–892.
[13] T.v. Dam, K. Langendoen, An adaptive energy-efficient MAC protocol for wireless sensor networks, in: Proceedings of the 1st international conference on Embedded networked sensor systems, ACM, Los Angeles, California, USA, 2003, pp. 171–180.
[14] T. Zheng, S. Radhakrishnan, V. Sarangan, PMAC: an adaptive energy-efficient MAC protocol for wireless sensor networks, in: Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05) – Workshop 12, vol. 13, IEEE Computer Society, 2005, pp. 237–231.
[15] G. Lu, B. Krishnamachari, C.S. Raghavendra, An adaptive energy-efficient and low-latency MAC for tree-based data gathering in sensor networks: research articles, Wireless Communications and Mobile Computing 7 (2007) 863–875.
[16] J. Polastre, J. Hill, D. Culler, Versatile low power media access for wireless sensor networks, in: Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, ACM, Baltimore, MD, USA, 2004, pp. 95–107.
[17] A. El-Hoiydi, J.-D. Decotignie, Low power downlink MAC protocols for infrastructure wireless sensor networks, Mobile Networks and Applications 10 (2005) 675–690.
[18] X. Chen, P. Han, Q.S. He, S.L. Tu, Z.L. Chen, A multi-channel MAC protocol for wireless sensor networks, in: Proceedings – Sixth IEEE International Conference on Computer and Information Technology, CIT 2006, Seoul, 2006, pp. 224–230.
[19] G. Zhou, C. Huango, T. Yan, T. He, J.A. Stankovic, T.F. Abdelzaher, MMSN: Multi-frequency media access control for wireless sensor networks, in: Proceedings – IEEE INFOCOM, Barcelona, 2006, pp. 1–13.
[20] Y. Kim, H. Shin, H. Cha, Y-MAC: An energy-efficient multi-channel MAC protocol for dense wireless sensor networks, in: Proceedings –

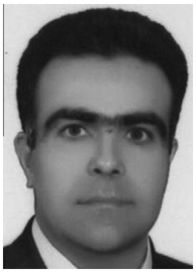
2008 International Conference on Information Processing in Sensor Networks, IPSN 2008, St. Louis, MO, 2008, pp. 53–63.
[21] H.K. Le, D. Henriksson, T. Abdelzaher, A practical multi-channel media access control protocol for wireless sensor networks, in: Proceedings – 2008 International Conference on Information Processing in Sensor Networks, IPSN 2008, St. Louis, MO, 2008, pp. 70–81.
[22] G.H. EkbataniFard, M.H. Yaghmaee, R. Monsefi, An adaptive cross-layer multichannel QoS-MAC protocol for cluster based wireless multimedia sensor networks, in: 2009 International Conference on Ultra Modern Telecommunications and Workshops, St. Petersburg, 2009, pp. 1–6.
[23] Y. Yang, W. Yi, Rainbow: Reliable data collecting MAC protocol for wireless sensor networks, in: IEEE Wireless Communications and Networking Conference, WCNC2010, Sydney, NSW, 2010, pp. 1–6.
[24] G. Zhou, Y. Wu, T. Yan, T. He, C. Huang, J.A. Stankovic, T.F. Abdelzaher, A multifrequency MAC specially designed for wireless sensor network applications, Transactions on Embedded Computing Systems 9 (2010).
[25] O.D. Incel, L. van Hoesel, P. Jansen, P. Havinga, MC-LMAC: a multi-channel MAC protocol for wireless sensor networks, Ad Hoc Networks 9 (2011) 73–94.
[26] C. Li, P. Wang, H.H. Chen, M. Guizani, A cluster based on-demand multi-channel MAC protocol for wireless multimedia sensor networks, in: IEEE International Conference on Communications, Beijing, 2008, pp. 2371–2376.
[27] M. Salajegheh, H. Soroush, A. Kalis, HYMAC: Hybrid TDMA/FDMA medium access control protocol for wireless sensor networks, in: IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC, Athens, 2007, pp. 1–5.
[28] Y. Wu, J.A. Stankovic, T. He, S. Lin, Realistic and efficient multi-channel communications in wireless sensor networks, in: Proceedings – IEEE INFOCOM, Phoenix, AZ, 2008, pp. 1867–1875.
[29] Y. Yang, Y. Liu, L.M. Ni, Level the buffer wall: fair channel assignment in wireless sensor networks, Computer Communications 33 (2010) 1370–1379.
[30] K.R. Chowdhury, N. Nandiraju, P. Chanda, D.P. Agrawal, Q.A. Zeng, Channel allocation and medium access control for wireless sensor networks, Ad Hoc Networks 7 (2009) 307–321.
[31] G. EkbataniFard, R. Monsefi, A detailed review of multi-channel medium access control protocols for wireless sensor networks, International Journal of Wireless Information Networks (2011) 1–21.
[32] J.R. Jiang, Y.C. Tseng, C.S. Hsu, T.H. Lai, Quorum-based asynchronous power-saving protocols for IEEE 802.11 ad hoc networks, Mobile Networks and Applications 10 (2005) 169–181.
[33] C.M. Chao, J.P. Sheu, I.C. Chou, An adaptive quorum-based energy conserving protocol for IEEE 802.11 ad hoc networks, IEEE Transactions on Mobile Computing 5 (2006) 560–570.
[34] S.M. Chen, S.P. Kuo, Y.C. Tseng, A quorum-based mechanism as an enhancement to clock synchronization protocols for IEEE 802.11 MANETs, IEEE Communications Letters 11 (2007) 313–315.
[35] J.R. Jiang, Expected quorum overlap sizes of quorum systems for asynchronous power-saving in mobile ad hoc networks, Computer Networks 52 (2008) 3296–3306.
[36] C.M. Chao, Y.Z. Wang, A multiple rendezvous multichannel MAC protocol for underwater sensor networks, in: IEEE Wireless Communications and Networking Conference, WCNC, Sydney, NSW, 2010, pp. 1–6.
[37] S.Y. Cheung, M.H. Ammar, M. Ahamad, The grid protocol: a high performance scheme for maintaining replicated data, IEEE Transactions on Knowledge and Data Engineering 4 (1992) 582–592.
[38] S.-D. Lang, L.-J. Mao, A torus quorum protocol for distributed mutual exclusion, in: Proceedings – 10th Intenational Conferences of Parallel and Distributed Computing System, 1998, pp. 635–638.
[39] W. Shan-Hung, C. Chung-Min, C. Ming-Syan, AAA: asynchronous, adaptive, and asymmetric power management for mobile ad hoc networks, in: INFOCOM 2009, IEEE, 2009, pp. 2541–2545.
[40] OPNET Modeler <http://www.opnet.com/solutions/network_rd/modeler.html>.
[41] T. Lei, S. Yanjun, O. Gurewitz, D.B. Johnson, PW-MAC: An energy-efficient predictive-wakeup MAC protocol for wireless sensor networks, in: INFOCOM, 2011 Proceedings IEEE, Shanghai, China, 2011, pp. 1305–1313.
[42] L. Tang, Y. Sun, O. Gurewitz, D.B. Johnson, EM-MAC: a dynamic multichannel energy-efficient MAC protocol for wireless sensor networks, in: MobiHoc, Paris, France, 2011, pp. 1–11.
[43] IEEE 802.15 wpan task group <http://www.ieee802.org/15/1735pub/TG4.html>, 2010.
[44] L.F.W.v. Hoesel, Sensors on Speaking Terms: Schedule-Based Medium Access Control Protocols for Wireless Sensor Networks, University of Twente, Enschede, 2007.
[45] Y. Gu, T. He, Data forwarding in extremely low duty-cycle sensor networks with unreliable communication links, in: Proceedings of the 5th International Conference on Embedded Networked Sensor Systems, ACM, Sydney, Australia, 2007, pp. 321–334.
[46] S. Guo, Y. Gu, B. Jiang, T. He, Opportunistic flooding in low-duty-cycle wireless sensor networks with unreliable links, in: Proceedings of the 15th Annual International Conference on Mobile Computing and Networking, ACM, Beijing, China, 2009, pp. 133–144.
[47] S. Ganeriwal, D. Ganesan, H. Sim, V. Tsiatsis, M. Srivastava, Estimating clock uncertainty for efficient duty-cycling in sensor networks, in: ACM SenSys., 2007.
[48] M.S.a.N.G. Shivaratri, Advanced Concepts in Operating Systems, McGraw-Hill, New York, 1994. p. 345.
[49] H. Garcia-Molina, D. Barbara, How to assign votes in a distributed system, Journal of ACM 32 (1985) 841–860.
[50] D. Agrawal, A.E. Abbadi, An efficient and fault-tolerant solution for distributed mutual exclusion, ACM Transactions on Computer Systems 9 (1991) 1–20.
[51] R.H. Thomas, A majority consensus approach to concurrency control for multiple copy databases, ACM Transactions on Database Systems 4 (1979) 180–209.
[52] Single-chip 2.4 Ghz IEEE 802.15.4 compliant and zigbee(tm) ready RF transceiver <http://www.ti.com/lit/gpn/cc2420>, 2010.
[53] MICAz datasheet <www.xbow.com>, 2011.

GholamHossein Ekbatanifard is currently pursuing his Ph.D. degree in Ferdowsi University of Mashhad, Khorasan Razavi, Iran. He obtained his B.Sc. degree in Software Engineering from the Islamic Azad University-Lahijan Branch, Guilan, Iran in 2001 and the M.Sc. degree in Computer Network Security from Ferdowsi University of Mashhad in 2004. He attends to Islamic Azad University from 2005. His research interests include design and performance evaluation of communication protocols for wireless sensor network and computer network security. He is a member of IEEE and Computer Society of Iran (CSI).



Reza Monsefi was born in 1956 in Ahwaz, the south of Iran. He has received his Honors Degree in Electrical and Electronic Engineering from Manchester University, Manchester, UK in 1978, M.Sc. degree in Control Engineering from Salford University, Manchester, in 1981, and Ph.D. in Data Communication and Supervisory Control, Salford University, Manchester in 1983. He is a fellow member of IEE, and at present a senior lecturer and cooperates with Ferdowsi University of Mashhad, Mashhad, Iran. Computer Networks, Wireless Sensor Networks, Machine Learning, and Soft Computing are among his professional interests.

Mohammad H. Yaghmaee M. was born on July 1971 in Mashhad, Iran. He received his B.S. degree in Communication Engineering from Sharif University of Technology, Tehran, Iran in 1993, and M.S. degree in communication engineering from Tehran Polytechnic (Amirkabir) University of Technology in 1995. He received his Ph.D. degree in communication engineering from Tehran Polytechnic (Amirkabir) University of Technology in 2000. He has been a computer network engineer with several networking projects in Iran Telecommunication Research Center (ITRC) since 1992. From November 1998 to July1999, he was with Network Technology Group (NTG), C& C Media research labs, NEC Corporation, Tokyo, Japan, as visiting research scholar. From September 2007 to August 2008, he was with the Lane Department of Computer Science and Electrical Engineering, West Virginia University, Morgantown, USA as a visiting associate professor. He is the author of 3 books all in Farsi language. He has published more than 85 international conference and journal papers. His research interests are in Wireless Sensor Networks (WSNs), traffic and congestion control, high-speed networks including ATM and MPLS, Quality of Services (QoSs) and fuzzy logic control. He is a senior member of IEEE.



Seyed Amin Hosseini S. received his B.Sc., and M.Sc., degree in Computer Engineering from Ferdowsi University of Mashhad, Iran in 1990 and 1998, respectively. He received his Ph.D. degree in Computer Network from Universiti Sains Malaysia in 2010. He is Dean of E-Learning Center, Ferdowsi University of Mashhad. His research interest includes wireless networks, energy efficient protocols, computer network protocols, and network security.