

## Research Article

# Dynamic Channel Slot Allocation Scheme and Performance Analysis of Cyclic Quorum Multichannel MAC Protocol

Xing Hu,<sup>1</sup> Linhua Ma,<sup>1</sup> Shaocheng Huang,<sup>2</sup> Tianyu Huang,<sup>1</sup> and Shiping Liu<sup>1</sup>

<sup>1</sup>School of Aeronautics and Astronautics Engineering, Air Force Engineering University, Xi'an, Shaanxi 710038, China

<sup>2</sup>School of Electrical Engineering, Royal Institute of Technology, Stockholm, Sweden

Correspondence should be addressed to Xing Hu; hustart@yahoo.com

Received 4 November 2016; Accepted 27 June 2017; Published 28 September 2017

Academic Editor: Ibrahim Zeid

Copyright © 2017 Xing Hu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In high diversity node situation, multichannel MAC protocol can improve the frequency efficiency, owing to fewer collisions compared with single-channel MAC protocol. And the performance of cyclic quorum-based multichannel (CQM) MAC protocol is outstanding. Based on cyclic quorum system and channel slot allocation, it can avoid the bottleneck that others suffered from and can be easily realized with only one transceiver. To obtain the accurate performance of CQM MAC protocol, a Markov chain model, which combines the channel-hopping strategy of CQM protocol and IEEE 802.11 distributed coordination function (DCF), is proposed. The results of numerical analysis show that the optimal performance of CQM protocol can be obtained in saturation bound situation. And then we obtain the saturation bound of CQM system by bird swarm algorithm. In addition, to improve the performance of CQM protocol in unsaturation situation, a dynamic channel slot allocation of CQM (DCQM) protocol is proposed, based on wavelet neural network. Finally, the performance of CQM protocol and DCQM protocol is simulated by Qualnet platform. And the simulation results show that the analytic and simulation results match very well; the DCQM performs better in unsaturation situation.

## 1. Introduction

IEEE 802.11 Distributed Coordination Function (DCF) [1] has been widely used in mobile ad hoc networks. However, in high diversity node situation, the IEEE 802.11 DCF suffers from many collisions, resulting in low frequency utilization [2].

Multichannel MAC protocol can help to share the traffic loads among different channels. Hence, the collisions single-channel MAC protocols suffered from can be alleviated. In fact, IEEE 802.11b and IEEE 802.11a can support 3 and 13 nonoverlapping channels, respectively [2]. This means that designing a multichannel MAC (MMAC) protocol is feasible and desirable in IEEE 802.11-based mobile ad hoc networks.

The great challenge of designing MMAC is how to allocate channels to node for transmitting information. And solutions to this problem can be classified into two patterns:

- (a) Channel negotiating pattern: channels allocated by negotiating through dedicated control slot or channel.
- (b) Channel-hopping pattern: nodes switch channels in a fixed pattern or random mechanism.

There are many protocols ([3–15]) that belong to the channel negotiating pattern. In these protocols, some are equipped with at least two transceivers, such as protocols [3, 7, 11–14]. For these protocols, one transceiver is turned to a common control channel that is dedicated for negotiating the channel to be used next. Once the channel is selected, the other transceiver(s) will switch to the negotiated channel for data transmission. These protocols are referred to as the multitransceiver channel negotiating solutions. An undesirable feature of these protocols is high hardware cost. In addition, at one moment, at most one transmission pair can be handled, and the dedicated control transceiver is a bottleneck of the throughput of network. To solve these problems, some are equipped with only one transceiver, such as [4–6, 8–10, 15]. Among them, protocols [5, 6, 9, 15] use a dedicated control channel for control message exchange. In such methods, the dedicated control channel will be either overloaded or underutilized if the capacity of the dedicated control channel and the data channels is not properly distributed. Some other proposals [4, 8, 10] use a common control period, which is similar to the ATIM window concept in IEEE

802.11 power-saving mode (PSM), for nodes to negotiate the data channels. Using either a dedicated control channel or a common control period, these protocols are referred to as the single-transceiver channel negotiating solutions. However, the bottleneck still exists, and no concurrent handshaking is allowed.

Different from the channel negotiating pattern, some other solutions utilize the channel-hopping concept to switch among data channels to achieve concurrent handshaking. Similarly, some of channel-hopping pattern protocols are equipped with at least two transceivers, such as protocols [16–18]. However, owing to the high hardware cost and difficulty to be realized in current equipment, we only discuss the single-transceiver protocols in the rest of paper. The protocols [19–21] are equipped with only one transceiver. In these protocols, different nodes have different channel-hopping sequences, and each pair of nodes is able to communicate with each other when they switch to the same channel at the same time slot. In [19], a multichannel protocol called multichannel MAC protocol (McMAC) uses a common linear congruential generator to build each node's channel-hopping sequence. An asynchronous efficient multichannel MAC protocol (EM-MAC) also adopts the pseudorandom channel-hopping mechanism in [20]. In the slotted seeded channel-hopping protocol (SSCH) [21], a channel-hopping mechanism is proposed such that any two nodes are guaranteed to have a rendezvous once in one cycle. A flaw of these channel-hopping protocols is that they may suffer from the missing receiver problem. This problem frequently occurs, particularly in a heavy-loaded environment [5]. In [2], Chao and Tsai propose a cyclic quorum-based multichannel (CQM) MAC protocol. According to channel-hopping strategy of CQM protocol, based on cyclic quorum system, every channel has the same chance to be accessed. Hence, it can overcome the bottleneck of channel negotiating pattern and the missing receiver problem of channel-hopping pattern.

However, as is known to us, the most optimal performance of CQM protocol is not analyzed theoretically

yet. And in unsaturation situation, the performance of the fixed channel slot allocation scheme cannot be optimal. Hence, to obtain the optimal performance, we analyze the performance of CQM protocol theoretically based on Markov chain model. And a traffic prediction based dynamic channel slot allocation scheme of CQM protocol is proposed.

The rest of this paper is organized as follows. The cyclic quorum system and CQM protocol are introduced in Section 2. The Markov chain model and performance analysis of CQM protocol are given in Section 3. The saturation bound is obtained by bird swarm optimal algorithm in Section 4. The traffic prediction based dynamic channel slot allocation scheme of CQM protocol is introduced in Section 5. In addition, the theoretical performance analysis CQM protocol is verified, and the performance comparison between CQM and DCQM protocols is also given in Section 6. The paper is concluded in Section 7.

## 2. Introduction of CQM Protocol

*2.1. Cyclic Quorum System.* Quorum systems have been widely used for MAC protocol designing in wireless networks [22, 23]. And a quorum system can be defined as follows.

Given a universal set  $U = \{0, \dots, n-1\}$ , a quorum system  $Q$  under  $U$  is a collection of nonempty subsets of  $U$ , each called a quorum, which satisfies the intersection property

$$G \cap H \neq \emptyset: \quad \forall G, H \in Q. \quad (1)$$

There are many quorum systems, such as the cyclic quorum system, the grid quorum system, and the torus quorum system. The cyclic quorum system has the elegant feature of rotation closure property, and it can provide equal opportunity for a node to transmit and to receive. The rotation closure property is as follows:

$$G \cap \text{rotate}(H, i) \neq \emptyset: \quad \forall G, H \in Q, i \in \{0, \dots, n-1\} \text{ where } \text{rotate}(H, i) = \{(j+i) \bmod n \mid j \in H\}. \quad (2)$$

A cyclic quorum system can be constructed from a difference set. The definitions of a difference set and a cyclic quorum system are as follows.

Given a set  $D = \{d_1, \dots, d_k\}$ ,  $d_k \in \mathbb{Z}_n$ , the set  $D$  can be a difference set if

$$\begin{aligned} & \forall (e \neq 0) \quad \bmod n, \\ & \exists (d_i - d_j) = e \quad \bmod n \text{ where } d_i, d_j \in D. \end{aligned} \quad (3)$$

Given a difference set  $D = \{d_1, \dots, d_k\}$ ,  $d_k \in \mathbb{Z}_n$ , a cyclic quorum  $Q$  can be defined as follows:

$$\begin{aligned} Q &= \{Q_0, \dots, Q_{n-1}\}, \\ \text{where } Q_i &= \{d_1 + i, \dots, d_k + i\} \bmod n, \quad i = 1, \dots, n. \end{aligned} \quad (4)$$

*2.2. CQM Protocol.* Based on cyclic quorum system, Chao and Tsai propose a cyclic quorum-based multichannel (CQM) MAC protocol [2]. The CQM protocol does not need channel negotiating, so the bottleneck of channel negotiating mechanism can be avoided. In CQM protocol, nodes can communicate with others when they meet at the same channel and same channel slot. And it can provide equal opportunity for nodes to transmit and to receive through the cyclic quorum system.

In CQM protocol, time is divided into a series of cycles. Each cycle consists of default channel slots and switching channel slots, numbered from 0 to  $z-1$ . The value of  $z$  is determined by the integer set from which the adopted difference set is derived. According to [2], CQM system

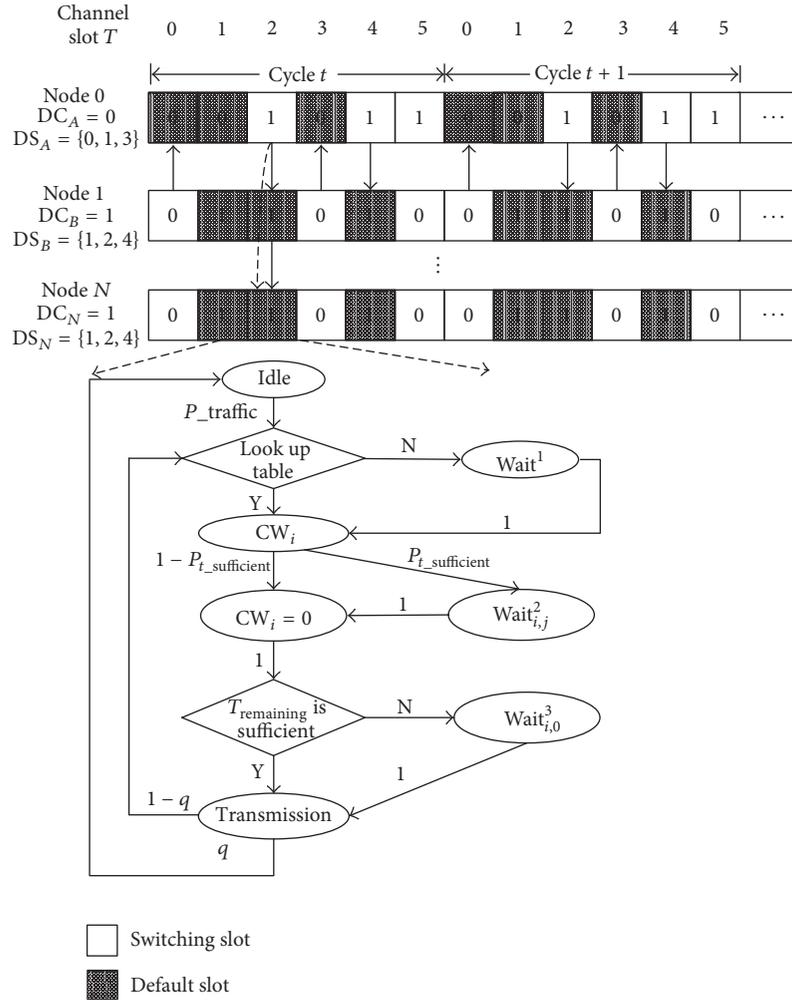


FIGURE 1: The protocol processing of CQM protocol with two channels.

performs better under  $Z_6$ . In this paper,  $Z_6$  is adopted in the analysis of CQM system.

Let  $T$  denote the channel slot length, to separate with system time slot  $\sigma$ . We assume that the length of channel slot  $T$  is long enough to transmit at least one data packet. At default channel slots, a node stays at its default channel, waiting for transmission requests. At switching channel slots, a node may switch to its intended receiver's default channel. We use a cyclic quorum  $G_i$  under  $Z_n$  to identify a node's default slots. Specifically, for any node  $i \in V$ , where  $V$  is the set of nodes in the network,  $i$ th node's default channel (denoted as  $DC_i$ ) and default slots (denoted as  $DS_i$ ) are chosen as follows:

$$\begin{aligned}
 DC_i &= \text{node\_ID}_i \pmod{h} \\
 DS_i &= G_j, \quad j = \text{node\_ID}_i \pmod{z}, \quad \text{where } i \in V,
 \end{aligned} \tag{5}$$

where  $\text{node\_ID}_i$  is the ID of  $i$ th node.

Figure 1 is an example of CQM operation under  $Z_6$  with two channels (numbered from 0 to 1). Nodes 0, 1, and  $N$ , with IDs 0, 1, and  $N$ , respectively, are within each other's transmission range. Each node's default channel and default

slots are shown in Figure 1, supposing that we choose the difference set  $\{0, 1, 3\}$  under  $Z_6$  as  $G_0$ . The marked time slots are default channel slots. The number in each channel slot is the channel that should be switched to.

As is shown in Figure 1, when a packet arrives at a random time slot, the packet should wait in queue until all preceding packets are transmitted. And then the node looks up its channel-hopping table and checks whether it can meet its destination node at the current channel slot. If they meet each other at the current channel slot, the node can turn into the back-off stage. Otherwise, the node will turn into  $\text{Wait}^1$  state to wait for the next channel slot available. During back-off stage, the back-off timer of a node is set to a random time according to 802.11 DCF scheme. In this paper, according to CQM protocol, when the channel is occupied by the other nodes or the current channel slot is empty, the back-off timer is assumed to be held to wait for the next channel slot available. And then, the channel slot remaining will be checked when back-off timer reaches zero. If the channel slot remaining is long enough for one packet transmission, the packet will be transmitted. Otherwise, the node should turn into  $\text{Wait}_{i,0}^3$  state to wait for the next channel slot available.

TABLE 1: Average channel slots analysis of node 0.

Quorum combinations	Waiting channel slots between two meetings		Channel slots in one cycle	
	Meeting state	Waiting state	Meeting slots	Waiting slots
$G_0 + G_0$	$1/h$	$4/3h$	$3/h$	$(6h - 3)/h$
$G_0 + G_1$	2	$7/4$	2	4
$G_0 + G_2$	2	2	2	4
$G_0 + G_3$	6	3	1	5
$G_0 + G_4$	3	$5/2$	2	4
$G_0 + G_5$	2	$3/2$	2	4
Results	$E(W\_meeting)$ $(15h + 1) / 6h$	$E(W\_waiting)$ $(129h + 16) / 12h$	$E(M\_slots)$ $(3h + 1) / 2h$	$E(W\_slots)$ $(9h - 1) / 2h$

Finally, after each successful transmission or after having reached the maximum number of the retransmissions  $m$ , if there is a packet to be transmitted in queue, the node enters into checking processing again. Otherwise, the node enters into the idle state.  $q$  denotes the probability of having an empty queue.

Take Node 0 for example; Table 1 shows average channel slots needed between twice meeting in one cycle in the situation that the node is at meeting state and waiting state. And the average meeting slots and waiting slots in one cycle are also shown in Table 1.

### 3. Performance Analysis of CQM Protocol

**3.1. Markov Model of CQM Protocol.** Combined with the channel-hopping strategy of CQM protocol and 802.11 DCF, the Markov chain model of CQM protocol is proposed, as is shown in Figure 2. We consider the following assumptions for modeling network. (1) The network consists of  $n$  nodes moving in the range of transmission distance, and packets are transmitted from sources to destinations directly. (2) A total of  $h$  channels are available for every node, and all of them have the same bandwidth. (3) Every node knows the IDs of its one-hop neighbors, and every node is equipped with one-half-duplex transceiver, which is able to switch to any channel dynamically. (4) Every node is time synchronized and uses the IEEE 802.11 DCF (RTS/CTS mode) as MAC protocol. (5) Packets arrive to a node according to a Poisson process with rate  $\lambda$  packets/s. (6) The channel is assumed to be error-free. Packet transmission is considered to be successful if there are no other packets transmissions at the same time.

According to CQM protocol, each node has the same chance to transmit packet through different channels. So we can take one selected channel of a node into consideration to model the CQM protocol scheme at MAC layer. And then we assume that there are  $n' = n/h$  nodes with packets rate  $\lambda$  to content in one selected channel in a random time slot. The model consists of states that a node can reside in. The points  $C_0, C_1, C_2, C'_0, \dots, C'_m$  represent connection points, which are not states.

During queuing stage, the arriving packet will wait in buffer, if there are other packets to be transmitted. And if there is no packet in buff, the node will turn into idle state.

If the packet is eligible to be transmitted, the node enters into checking stage. The channel-hopping table will be checked firstly. The node can meet its destination node with probability  $P_{meeting}$  and turns into Wait<sup>1</sup> state to wait for next channel slot available with probability  $1 - P_{meeting}$ .

During back-off stage, we use the tuple  $(i, k)$  to represent the different states.  $i$  denotes the  $i$ th back-off stage, and number  $i = 0, 1, \dots, m', \dots, m$ , and  $k$  denotes the value of back-off timer in the range  $[0, W_i - 1]$  [24].  $W_i$  is the size of the CW at stage  $i$ , which can be expressed as follow:

$$W_i = \begin{cases} 2^i * W_0, & \text{if } i \leq m' \\ 2^{m'} * W_0, & \text{if } i > m', \end{cases} \quad (6)$$

where  $m$  denotes the maximum number of packet retransmissions before the packet is dropped. According to [25], the default value for  $m'$  is 5 and it is 7 for  $m$ .

When time slot  $\sigma$  is idle and the current channel slot remaining is sufficient for once back-off timer decrease, the state  $(i, k)$  will turn into  $(i, k - 1)$  with probability  $P'_{t\_sufficient}$ . Otherwise, the back-off timer will be held and the node will wait for next channel slot available. According to 802.11 DCF, the packet will be transmitted when node turns into  $(i, 0)$  states. Different from that, in this model, only if channel slot remaining is long enough for one packet transmission, the packet will be transmitted at  $(i, 0)$  states. Otherwise, the node will enter into Wait<sup>3</sup> <sub>$i,0$</sub>  state to wait for next channel slot available.  $P_{i,k}$  denotes the probability of state  $(i, k)$ .  $P_{t\_success}$  denotes the probability that a node transmits successfully in a random time slot and one selected channel. We use  $P_{C_i}$  to denote the probability that the node arrives at point  $C_i$ .

In order to obtain the expression of state-steady probabilities of a node, firstly, we need to express  $P_{i,k}$  in terms of  $P_{0,0}$ . According to Figure 2, during back-off stage, we can obtain the following relations:

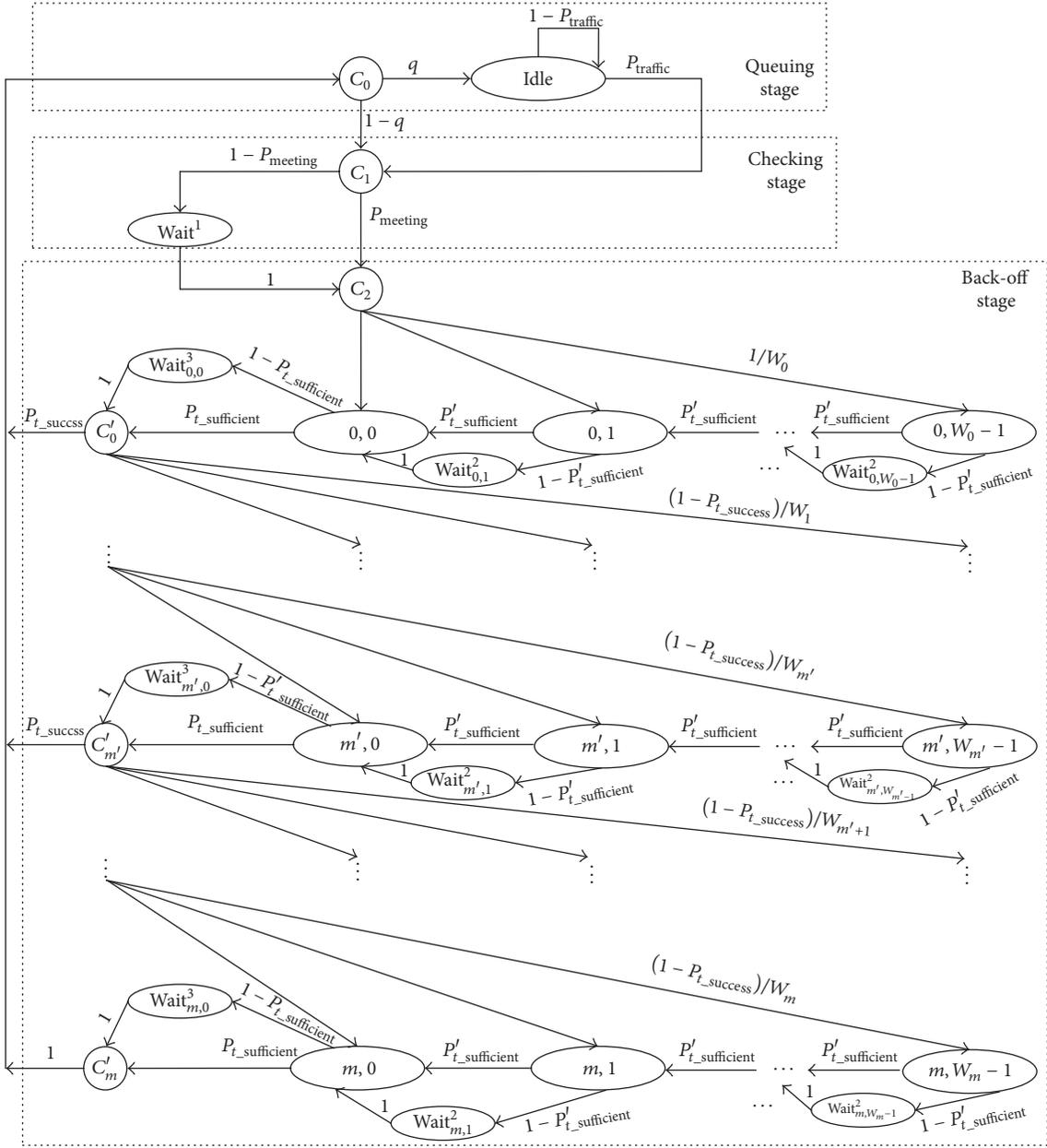


FIGURE 2: Markov chain model of CQM protocol.

$$P_{C_i} = P_{i,0} * P_{t\_sufficient} + P_{Wait^3_{i,0}} = P_{i,0} * P_{t\_sufficient} + P_{i,0} * (1 - P_{t\_sufficient}) = P_{i,0}, \quad 0 \leq i \leq m,$$

$$P_{i,k} = P_{i,k+1} * P'_{t\_sufficient} + P_{Wait^2_{i,k}} = P_{i,k+1} * P'_{t\_sufficient} + P_{i,k+1} * (1 - P'_{t\_sufficient}) = P_{i,k+1}, \quad 0 \leq j \leq W_i - 1,$$

$$P_{i,0} = \begin{cases} \sum_{u=0}^{w_i-1} P_{C_{i-1}} * \frac{1 - P_{t\_success}}{w_i} = (1 - P_{t\_success})^i * P_{0,0}, & 1 \leq i \leq m, \\ \sum_{u=0}^{w_0-1} P_{C_2} * \frac{1}{w_0} = P_{C_2}, & i = 0, \end{cases}$$

$$P_{i,k} = \begin{cases} \sum_{u=0}^{w_i-k-1} P_{C'_{i-1}} * \frac{1 - P_{t\_success}}{w_i} = \frac{w_i - k}{w_i} * (1 - P_{t\_success})^i * P_{0,0}, & 1 \leq i \leq m, \\ \sum_{u=0}^{w_0-k-1} P_{C_2} * \frac{1}{w_0} = \frac{w_0 - k}{w_0} * P_{C_2} = \frac{w_0 - k}{w_0} * P_{0,0}, & i = 0, \end{cases} \quad (7)$$

where  $P_{t\_sufficient}$  and  $P'_{t\_sufficient}$  denote the probability that the remaining channel time is enough for one packet transmission and once back-off timer decrease, respectively. They can be expressed as follows:

$$P_{t\_sufficient} = \frac{T - T_s}{T}, \quad (8)$$

$$P'_{t\_sufficient} = \frac{T - \sigma}{T}.$$

Let  $P_r^s$  denote the probability that only  $s$  nodes out of  $r$  nodes transmit packets in a random time slot and one selected channel. Hence,  $P_{t\_success}$  can be expressed as follows:

$$P_{t\_success} = P_{n'-1}^0 = C_{n'-1}^0 (\tau)^0 (1 - \tau)^{n'-1} = (1 - \tau)^{n'-1}, \quad (9)$$

where  $\tau$  denotes the probability that a node transmits one packet in a random time slot and one selected channel.  $\tau$  can be expressed as follows:

$$\tau = \sum_{i=0}^m P_{C'_i} = \sum_{i=0}^m (1 - P_{t\_success})^i * P_{0,0} = \alpha * P_{0,0}, \quad (10)$$

where  $\alpha = \frac{1 - (1 - P_{t\_success})^{m+1}}{P_{t\_success}}$ .

Combining (9) with (10),  $P_{0,0}$  can be expressed with  $P_{t\_success}$ :

$$P_{0,0} = \frac{1 - P_{t\_success}^{1/(n'-1)}}{\alpha}. \quad (11)$$

And then, the  $P_{C_i}$  can be obtained as follows:

$$P_{C_0} = \sum_{i=0}^{m-1} P_{C'_i} * P_{t\_success} + P_{C'_m}$$

$$= \sum_{i=0}^{m-1} (1 - P_{t\_success})^i * P_{0,0} * P_{t\_success} + (1 - P_{t\_success})^m * P_{0,0} = P_{0,0}, \quad (12)$$

$$P_{Idle} = \frac{P_{C_0} * q}{P_{traffic}} = \frac{q}{P_{traffic}} P_{0,0}, \quad (13)$$

$$P_{C_1} = P_{C_0} * (1 - q) + P_{Idle} * P_{traffic} = P_{C_0} = P_{0,0}, \quad (14)$$

$$P_{C_2} = P_{C_1} * P_{meeting} + P_{Wait1}$$

$$= P_{C_1} * P_{meeting} + P_{C_1} * (1 - P_{meeting}) = P_{C_1} \quad (15)$$

$$= P_{0,0},$$

where  $P_{meeting}$  denotes the probability that a pair of nodes can meet each other in a random channel slot and a selected channel. According to analysis above,  $P_{meeting}$  can be expressed as follows:

$$P_{meeting} = \frac{E(M\_slots)}{6} = \frac{3h + 1}{12h}. \quad (16)$$

Let  $P_{Wait^1}$ ,  $P_{Wait^2_{i,k}}$ , and  $P_{Wait^3_{i,0}}$  denote the probability of  $Wait^1$ ,  $Wait^2_{i,k}$ , and  $Wait^3_{i,0}$  states, respectively.  $T_{Wait^1}$ ,  $T_{Wait^2_{i,k}}$ , and  $T_{Wait^3_{i,0}}$  denote the time cost at  $Wait^1$ ,  $Wait^2_{i,k}$ , and  $Wait^3_{i,0}$  states. According to Figure 2, they can be expressed as follows:

$$P_{Wait^1} = P_{C_1} * (1 - P_{meeting}) = \frac{9h - 1}{12h} * P_{0,0},$$

$$T_{Wait^1} = E(W\_waiting) * T = \frac{129h + 16}{12h} * T,$$

$$P_{Wait^2_{i,k}} = \frac{(W_i - k) * (1 - P_{t\_success})^i * \sigma * P_{0,0}}{W_i T},$$

$$T_{Wait^2_{i,k}} = E(W\_meeting) * T = \frac{(15h + 1) * T}{6h}, \quad (17)$$

$$P_{Wait^3_{i,0}} = \frac{(1 - P_{t\_success})^i * P_{0,0} * T_s}{T},$$

$$T_{Wait^3_{i,0}} = \frac{T_s}{2} + E(W\_meeting) * T$$

$$= \frac{(15h + 1) * T + 3hT_s}{6h},$$

where  $P_{traffic} = 1 - e^{-\lambda\sigma}$  denotes the probability of having packet to be transmitted in a random time slot  $\sigma$  when the node is at Idle state.  $q$  denotes the probability that there is no packet in the queue of a node after each successful transmission or after having reached the maximum number of the retransmissions  $m$ .  $q$  can be expressed as follows:

$$q = e^{-\lambda * E[S_b]}. \quad (18)$$

And then (13) can be rewritten as follows:

$$P_{Idle} = \frac{q}{P_{traffic}} P_{0,0} = \frac{q}{1 - e^{-\lambda\sigma}} P_{0,0}, \quad (19)$$

where  $E[S_b]$  denotes the average service time of one packet transmission (from the packet leaving the MAC buffer until it is successfully transmitted or reaches the retry limit). According to Figure 2,  $E[S_b]$  includes two section: the first one is the

average time cost in checking stage ( $E[T_{\text{Wait}^1}]$ ), and the other is the average time cost in back-off stage ( $E[T_{\text{backoff}}]$ ):

$$E[S_b] = E[T_{\text{Wait}^1}] + E[T_{\text{backoff}}]. \quad (20)$$

According to Figure 2,  $E[T_{\text{Wait}^1}]$  can be expressed as follows:

$$\begin{aligned} E[T_{\text{Wait}^1}] &= T_{\text{Wait}^1} * P_{\text{Wait}^1} \\ &= \frac{(9h-1) * (129h+16) * T}{144h^2} * P_{0,0}. \end{aligned} \quad (21)$$

To get the value of all states in Figure 2, the average time cost  $E[T_{\text{backoff}}]$  in back-off stage should be obtained firstly. According to Figure 2,  $E[T_{\text{backoff}}]$  can be expressed as follows:

$$\begin{aligned} E[T_{\text{backoff}}] &= \bar{\sigma} \sum_{i=0}^m (1 - P_{t\_success})^i \frac{W_i - 1}{2} + \sum_{i=0}^m \left( T_S \right. \\ &\quad \left. + T_{\text{Wait}_{i,0}^3} * (1 - P_{t\_succicent}) \right) (1 - P_{t\_success})^i \\ &\quad \cdot P_{t\_success} + \sum_{i=1}^m \left( T_C + T_{\text{Wait}_{i,0}^3} * (1 - P_{t\_succicent}) \right) \\ &\quad * \left( (1 - P_{t\_success})^i P_{t\_success}^i \right. \\ &\quad \left. + (m+1)(1 - P_{t\_success})^{m+1} \right). \end{aligned} \quad (22)$$

Combining (6) with (22),  $E[T_{\text{backoff}}]$  can be expressed with  $P_{t\_success}$

$$\begin{aligned} E[T_{\text{backoff}}] &= \frac{\bar{\sigma}}{2} \left( \frac{(1 - (2(1 - P_{t\_success}))^{m+1}) W_0}{1 - 2(1 - P_{t\_success})} \right. \\ &\quad \left. + \frac{2^{m'} (1 - P_{t\_success})^{m'+1} W_0 (1 - (1 - P_{t\_success})^{m-m'})}{P_{t\_success}} \right. \\ &\quad \left. - \alpha \right) + \left( T_S + \frac{(15h+1) * T * T_S + 3hT_S^2}{6hT} \right) \alpha \\ &\quad * P_{t\_success} + \left( T_C + \frac{(15h+1) * T * T_S + 3hT_S^2}{6hT} \right) \\ &\quad * (1 - P_{t\_success}) * \alpha. \end{aligned} \quad (23)$$

Assume

$$\begin{aligned} \beta &= \frac{(1 - (2(1 - P_{t\_success}))^{m+1}) W_0}{1 - 2(1 - P_{t\_success})} \\ &\quad + \frac{2^{m'} (1 - P_{t\_success})^{m'+1} W_0 (1 - (1 - P_{t\_success})^{m-m'})}{P_{t\_success}}, \quad (24) \\ \gamma &= \frac{(15h+1) * T * T_S + 3hT_S^2}{6hT}, \end{aligned}$$

and then (23) can be rewritten as follows:

$$\begin{aligned} E[T_{\text{backoff}}] &= \frac{\bar{\sigma}}{2} * (\beta - \alpha) + (T_S + \gamma) * \alpha * P_{t\_success} \\ &\quad + (T_C + \gamma) * (1 - P_{t\_success}) * \alpha, \end{aligned} \quad (25)$$

where  $\bar{\sigma}$  denotes the average time between successive counter decrements in back-off stage. The time needed for successive counter decrements includes situations: only one time slot  $\sigma'$  is needed when all the other nodes have no packet to be transmitted,  $T_s + \sigma'$  needed when a packet is transmitted successfully by the other nodes, and  $T_c + \sigma'$  needed when a transmission collision occurs in the other nodes. Let  $P_{\sigma'}$ ,  $P_{T_s+\sigma'}$ , and  $P_{T_c+\sigma'}$  denote probabilities of these situations, respectively. Where  $\sigma'$  denotes the average time cost for once back-off timer decrease when the channel is idle.  $\sigma'$  can be expressed as follows:

$$\begin{aligned} \sigma' &= \sigma * P'_{t\_sufficient} + T_{\text{Wait}_{i,k}^2} * (1 - P'_{t\_sufficient}) \\ &= \frac{(T - \sigma) \sigma}{T} + \frac{(15h+1) \sigma}{6h}. \end{aligned} \quad (26)$$

And then, we can get

$$\begin{aligned} P_{\sigma'} &= P_{n'-1}^0 = (1 - \tau)^{n'-1}, \\ P_{T_s+\sigma'} &= P_{n'-1}^1 = C_{n'-1}^1 \tau (1 - \tau)^{n'-2} \\ &= (n' - 1) \tau (1 - \tau)^{n'-2}, \\ P_{T_c+\sigma'} &= 1 - P_{\sigma'} - P_{T_s+\sigma'} \\ &= 1 - (1 - \tau)^{n'-1} - (n' - 1) \tau (1 - \tau)^{n'-2}. \end{aligned} \quad (27)$$

$\bar{\sigma}$  can be expressed as follows:

$$\begin{aligned} \bar{\sigma} &= \sigma' * P_{\sigma'} + (T_s + \sigma') * P_{T_s+\sigma'} + (T_c + \sigma') \\ &\quad * P_{T_c+\sigma'}, \end{aligned} \quad (28)$$

where  $T_s$  is the time cost for a successful transmission and  $T_c$  is the average time the channel is sensed busy by each station during a collision. Let  $H = \text{PHY}_{hdr} + \text{MAC}_{hdr}$  be the packet header, and let  $\delta$  be the propagation delay.  $T_s$  and  $T_c$  are equal to

$$\begin{aligned} T_s &= \text{DIFS} + \text{RTS} + \text{CTS} + H + E[P] + \text{ACK} \\ &\quad + 3\text{SIFS} + 4\delta, \end{aligned} \quad (29)$$

$$T_c = \text{DIFS} + \text{RTS} + \text{CTS} + \text{SIFS} + \delta.$$

Based on this analysis, the value of  $q$  and  $P_{t,\text{success}}$  should be obtained before all the state-steady probabilities of a node can be got. Firstly, the relation between them can be expressed based on (18) and (20):

$$q = e^{-\lambda' * E[S_b]} = e^{-\lambda' * (E[T_{\text{wait}^1}] + E[T_{\text{backoff}}])}. \quad (30)$$

In addition, the relation between  $q$  and  $P_{t,\text{success}}$  also can be obtained by using the normalization condition

$$\begin{aligned} & \sum_{i=0}^m \sum_{k=0}^{w_i-1} P_{i,k} + P_{\text{Wait}^1} + P_{\text{Idle}} + \sum_{i=0}^m \sum_{k=1}^{w_i-1} P_{\text{Wait}_{i,k}^2} + \sum_{i=0}^m P_{\text{Wait}_{i,0}^3} \\ & = 1, \end{aligned}$$

$$P_{0,0} = \frac{1}{(1 + \sigma/T) * (\beta - \alpha) / 2 + (9h - 1) / 12h + q / (1 - e^{-\lambda\sigma}) + (\alpha * T_s) / T}. \quad (32)$$

Equations (30) and (32) are nonlinear and nonconvex. The equation set can be solved using arithmetic solution. And then, all the probabilities of  $P_{i,k}$ ,  $P_{C_i}$ ,  $P_{\text{Idle}}$ ,  $P_{\text{Wait}^1}$ ,  $P_{\text{Wait}_{i,k}^2}$ , and  $P_{\text{Wait}_{i,0}^3}$  can be got too.

**3.2. Throughput Analysis.** Now, we will analyze the throughput of CQM protocol. Let  $S$  denote the throughput of CQM system. In this paper, we assume the total of  $h$  channels is available, and all of them have the same bandwidth. Hence, the throughput of CQM protocol can be expressed as follows:

$$S = \sum_{i=1}^h \bar{S} = h * \frac{E[l]}{E[\text{slot}]}, \quad (33)$$

where  $E[l]$  denotes the average payload size per virtual slot in one selected channel.  $E[l]$  can be expressed as follows:

$$\begin{aligned} E[l] &= P_n^1 * l = C_n^1 \tau (1 - \tau)^{n'-1} * l \\ &= n' \tau (1 - \tau)^{n'-1} * l. \end{aligned} \quad (34)$$

And  $E[\text{slot}]$  denotes the virtual slot duration. It contains the duration of an empty slot time  $\sigma$ , the time duration  $T_S$  due to a successful transmission, and the time duration  $T_C$  due to a collision, respectively. And then the average slot duration  $E[\text{slot}]$  can be obtained:

$$E[\text{slot}] = P_n^0 * \sigma + P_n^1 * T_S + (1 - P_n^0 - P_n^1) * T_C. \quad (35)$$

Finally, the normalized throughput of CQM protocol can be obtained:

$$\begin{aligned} \bar{S} &= \frac{S}{h * C} \\ &= \frac{P_n^1 * l}{(P_n^0 * \sigma + P_n^1 * T_S + (1 - P_n^0 - P_n^1) * T_C) * C}. \end{aligned} \quad (36)$$

$$\begin{aligned} & \left(1 + \frac{\sigma}{T}\right) * \frac{\beta - \alpha}{2} * P_{0,0} + \frac{9h - 1}{12h} * P_{0,0} + \frac{q}{1 - e^{-\lambda\sigma}} \\ & * P_{0,0} + \frac{\alpha * T_s}{T} * P_{0,0} = 1, \\ & P_{0,0} * \left( \left(1 + \frac{\sigma}{T}\right) * \frac{\beta - \alpha}{2} + \frac{9h - 1}{12h} + \frac{q}{1 - e^{-\lambda\sigma}} \right. \\ & \left. + \frac{\alpha * T_s}{T} \right) = 1. \end{aligned} \quad (31)$$

The equation can be expressed as follows:

**3.3. Average Packet Transmission Delay Analysis.** In this section, we will analyze the expected packet transmission delay at the MAC layer of CQM protocol. Our analysis considers the delay  $E[D_{\text{tr}}]$  for a successfully transmitted packet in the case of saturation.  $E[D_{\text{tr}}]$  is defined as the time when a packet becomes eligible to be transmitted until it is transmitted successfully. So,  $E[D_{\text{tr}}]$  is equal to the back-off service time ( $E[S_b]$ ) excluding the term corresponding to dropping a packet ( $E[T_{\text{drop}}]$ ):

$$\begin{aligned} E[D_{\text{tr}}] &= E[S_b] - E[T_{\text{drop}}] \\ &= E[T_{\text{wait}^1}] + E[T_{\text{backoff}}] - (1 - P_{t,\text{success}})^{m+1} \\ & * \left( (m + 1) * (T_c + \gamma) + \bar{\sigma} * \sum_{i=0}^m \frac{W_i - 1}{2} \right). \end{aligned} \quad (37)$$

**3.4. Numerical Analysis of CQM Protocol.** To show the influence of parameters ( $n$ ,  $\lambda$ ,  $T$ , and  $h$ ) to the performance ( $E[D_{\text{tr}}]$  and  $S$ ) of CQM system obviously, we now present the relationship between them by numerical analysis when  $h$  is set to 3. A summary of parameters is shown in Table 2.

The throughput versus node number and packet rate is shown in Figure 3, when the channel slot length is set to 100 ms.

As is shown in Figure 3(a), the throughput increases gradually with the increased node number with a certain packet rate, and it approaches a constant in the end. In a network, with the node number increases, more packets can be transmitted successfully, so the throughput of network increases. However, with the network size become much bigger, more and more nodes transmit packets in one channel with a certain bandwidth, more and more collisions will occur. In the end, the network will reach a balance state, and the throughput will approach a constant. As is similar to the influence of node number, the throughput has the same change trend with the increased packet rate.

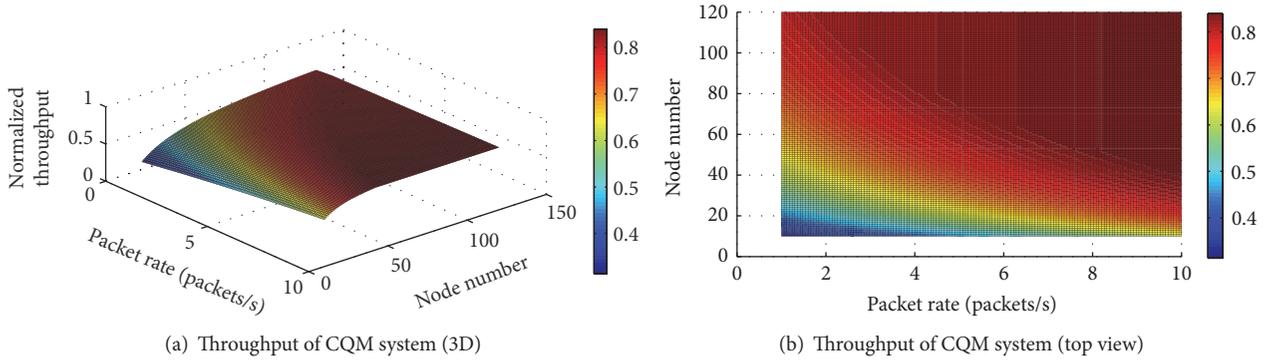


FIGURE 3: Throughput versus node number and packet rate with  $T = 100$  ms.

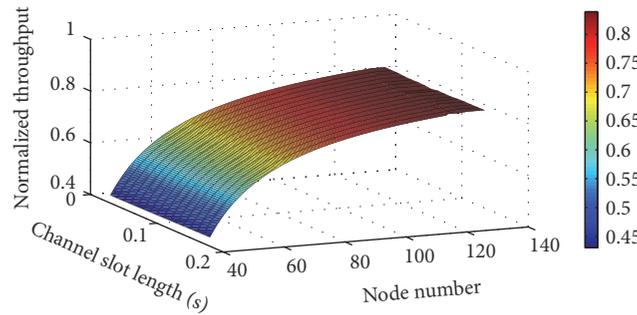


FIGURE 4: Throughput versus node number and channel slot length with  $\lambda = 2$ .

TABLE 2: CQM system parameters in numerical analysis.

Packet length	8192 bits
MAC header	272 bits
Physical header (PHY)	128 bits
ACK	112 bits + PHY header
RTS	160 bits + PHY header
CTS	112 bits + PHY header
SIFS	28 us
DIFS	128 us
$m$	6
$m'$	5
Slot duration	50 us
Propagation delay	1 us
Channel bit rate	1 Mbps
Channel number	3

As is shown in Figure 3(b), with the increase of node number and packet rate, the throughput increases gradually. When node number and packet rate reach a bound, the network will turn into saturation state, and throughput reaches its maximum value. In this paper, we call it saturation bound.

Figure 4 shows the relationship between throughput, node number, and channel slot length. This system consists of 3 channels, and the packet rate is set to 2 packets/s.

As is shown in Figure 4, on the one hand, the throughput increases with the increase of node number in a certain channel slot length, and it approaches a constant value when the network is at saturation state. On the other hand, the throughput changes slowly with the increase of channel slot length in a certain node number.

The relation between parameters  $(n, \lambda, T)$  and  $S$  can be revealed by numerical analysis. The most maximum throughput can be obtained when parameters  $(n, \lambda, T)$  break through the saturation bound. And then, the change trend of average packet delay with the change of parameters  $(n, \lambda, T)$  can be shown in Figures 5(a) and 5(b).

As is shown in Figure 5(a), with a given  $T = 100$  ms, the average packet delay increases sharply with the increase of node number in oversaturation case. The increased packet rate has little influence on the average packet delay. However, the packet dropping rate will increase sharply with the increase of packet rate in oversaturation case. Figure 5(b) shows that, with the given  $\lambda = 8$  and  $n = 36$ , the maximum throughput and minimum average packet transmission delay can be obtained when channel slot length is set to 0.1s. That is to say, for a given  $\lambda$  and  $n$ , the network can reach the saturation situation only when the channel slot length reaches one certain value.

Based on the analysis in this section, for a network with a certain channel slot length, the maximum throughput, minimum average packet delay, and minimum drop rate can be obtained only in the saturation bound case. And we will

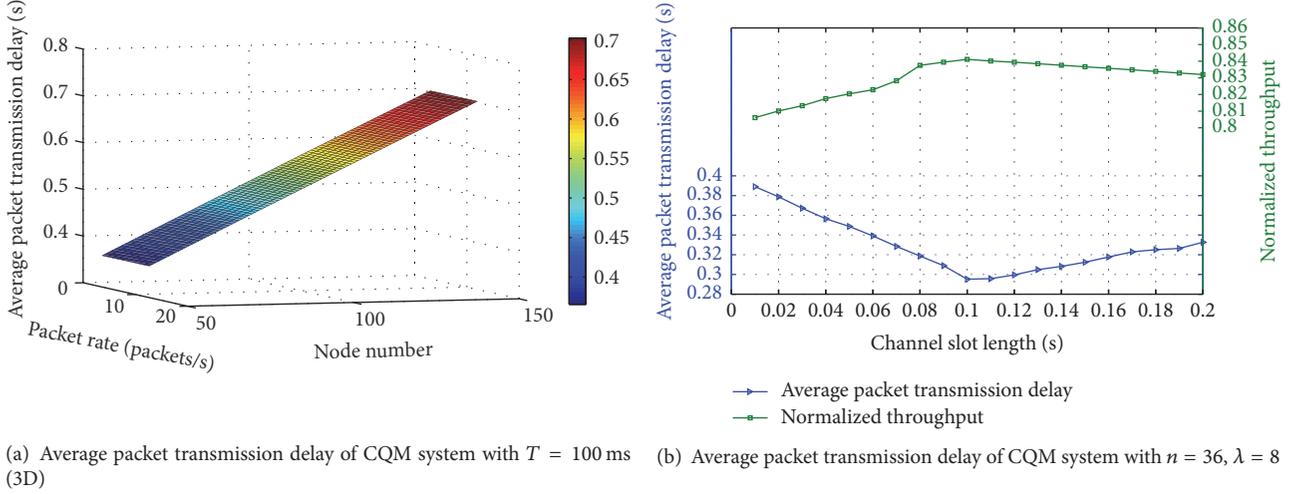


FIGURE 5: Average packet transmission delay in oversaturation case.

find the saturation bound of CQM system based on bird swarm algorithm in Section 5.

#### 4. Bird Swarm Algorithm Based Saturation Bound

In this section, we will find the saturation bound of CQM system. According to analysis in Section 3, the best performance of CQM system can be obtained when the network is in saturation bound case. In this way, the problem can be considered to be an optimization problem. The goal is to find pairs of value of  $n$  and  $\lambda$  in saturation bound case with a certain  $T$ .

For this optimization problem, to get the less average packet transmission delay and less drop rate under the situation with the most maximum throughput, we set the objective function as follows:

$$F = \bar{S} * 1000 + E[D_{tr}] + \lambda. \quad (38)$$

And the constraint conditions are as follows:

$$\begin{aligned} n &\in \mathbb{Z}; \\ \lambda &> 0 \\ 0 &< P_{t,\text{success}} \leq 1; \\ 0 &< q \leq 1. \end{aligned} \quad (39)$$

According to (38) and (39), this optimization problem is nonlinear and nonconvex. To attack this optimization problem, bird swarm algorithm (BSA) is adopted.

BSA has elegant properties of effectiveness, superiority, and stability to solve nonlinear and nonconvex problem, proposed by Meng et al. in 2015 [26]. It is based on the three kinds of behaviors of bird: foraging behavior, vigilance

behavior, and flight behavior. The BSA concludes four search strategies associated with five simplified rules.

(a) *Foraging Behavior*. Each bird searches for food according to its experience and the swarms' experience. This behavior can be expressed as follows:

$$\begin{aligned} x_{i,j}^{t+1} &= x_{i,j}^t + (p_{i,j} - x_{i,j}^t) \times \text{Cog} \times \text{rand}(0, 1) \\ &+ (g_j - x_{i,j}^t) \times \text{Soc} \times \text{rand}(0, 1), \end{aligned} \quad (40)$$

where  $x_{i,j}^t$  denotes the position of the  $i$ th bird at time step  $t$ .  $\text{rand}(0, 1)$  denotes independent uniformly distributed numbers in  $(0,1)$ .  $\text{Cog}$  and  $\text{Soc}$  are two positive numbers, which can be, respectively, called cognitive and social accelerated coefficients.  $p_{i,j}$  denotes the best previous position of the  $i$ th bird and  $g_j$  denotes the best previous position shared by the swarm.

(b) *Vigilance Behavior*. Birds would try to move to the centre of swarm, and they would inevitably compete with each other. Thus, each bird would not directly move towards the centre of the swarm. These motions can be formulated as follows:

$$\begin{aligned} x_{i,j}^{t+1} &= x_{i,j}^t + A1 (\text{mean}_j - x_{i,j}^t) \times \text{rand}(0, 1) \\ &+ A2 (p_{k,j} - x_{i,j}^t) \times \text{rand}(-1, 1), \\ A1 &= a1 \times \exp\left(-\frac{p\text{Fit}_i}{\text{sumFit} + \varepsilon} \times N\right), \\ A2 &= a2 \end{aligned} \quad (41)$$

$\times \exp\left(\left(\frac{p\text{Fit}_i - p\text{Fit}_k}{|p\text{Fit}_k - p\text{Fit}_i| + \varepsilon}\right) \frac{p\text{Fit}_i \times N}{\text{sumFit} + \varepsilon}\right)$ , where  $k$  is a positive integer, which is randomly chosen between 1 and  $N$ .  $a1$  and  $a2$  are two positive constants in  $[0, 2]$ .  $p\text{Fit}_i$  denotes the  $i$ th bird's best fitness value and  $\text{sumFit}$  represents the sum of the swarms' best fitness value.  $\text{mean}_j$

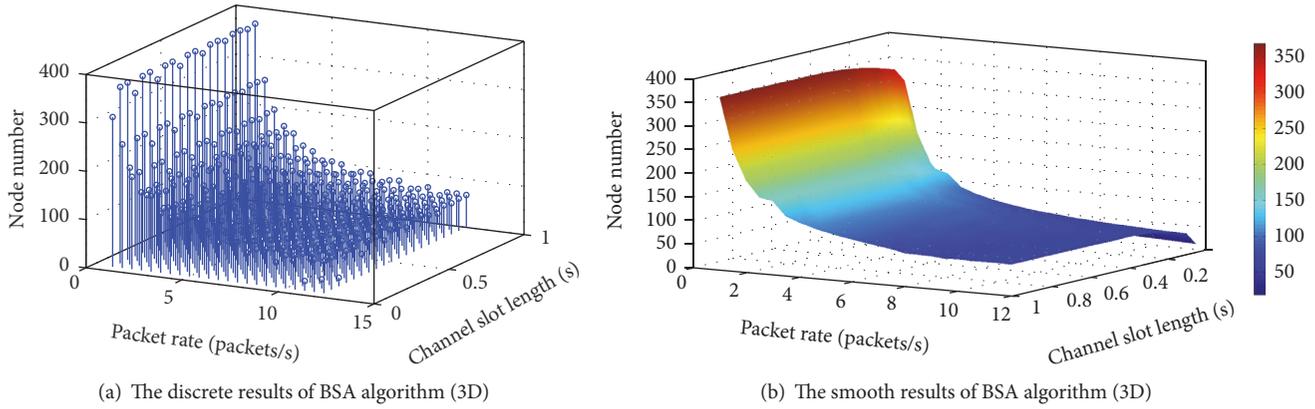


FIGURE 6: The saturation bound results obtained by BSA algorithm.

denotes the  $j$ th element of the average position of the whole swarm. And  $\varepsilon$  is the smallest constant in the computer.

(c) *Flight Behavior.* Birds may fly to another site in response to predation threat, foraging, or any other reasons. The behaviors of the producers and scroungers can be described mathematically as follows, respectively:

$$\begin{aligned} x_{i,j}^{t+1} &= x_{i,j}^t + \text{rand}n(0,1) \times x_{i,j}^t, \\ x_{i,j}^{t+1} &= x_{i,j}^t + (x_{k,j}^t - x_{i,j}^t) \times \text{FL} \times \text{rand}(0,1), \end{aligned} \quad (42)$$

where  $\text{rand}n(0,1)$  denotes Gaussian distributed random number with mean 0 and standard deviation 1. FL means that the scrounger would follow the producer to search for food.

Based on BSA algorithm, for a certain channel slot length  $T$ , we can get the tuple  $(n, \lambda)$  at the saturation bound. The parameter values of BSA algorithm in this section are given in Table 3.

In this section, the channel slot length  $T$  is selected in range of  $[0.01, 1]$ . And Figure 6(a) shows the results of BSA algorithm with 3 channels. Figure 6(b) is the smooth result of Figure 6(a).

The surface in Figure 6(b) can denote the saturation bound of CQM system obviously. The point on the surface denotes one possible tuple  $(n, \lambda, T)$  in the saturation bound case. As is shown in Figure 6(b), the packet rate changes against the changes of node number with a certain channel slot.

When  $T$  is set to 100 ms, the relationship between node number and packet rate for different channel number in the saturation bound case can be illustrated in Figure 7. And the normalized throughput for different node number with  $\lambda = 8$  can be shown in Figure 8.

As is shown in Figure 7, in saturation bound case, the node number decreases with the increase of packet rate with  $T = 100$  ms for different channel number. And the node number needed increase with the increase of channel number in a certain packet rate. That is to say, only when the system load is larger than capacity of channel, the network can reach the saturation bound state.

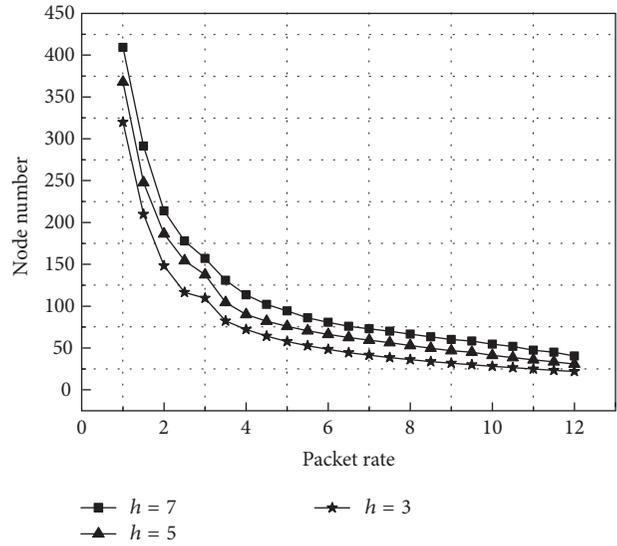


FIGURE 7: The saturation bound of CQM system with different channel number.

## 5. Traffic Prediction Based DCQM Protocol

According to the analysis in Section 4, the accurate performance of CQM MAC protocol is obtained. The metrics (throughput and average packet transmission delay) are calculated in performance analysis, with respect to node number, packet rate, channel slot length, and channel number.

In CQM protocol, owing to its equal opportunity for nodes to transmit and to receive, the optimal performance of CQM protocol can be obtained in saturation bound situation.

However, in unsaturation situation, in a random time slot, some nodes have packets to be transmitted, while others have no packet to be transmitted. In this situation, the channel slots can not be utilized effectively, due to the fixed slot allocation scheme of CQM protocol.

To get the optimal performance in both unsaturation and saturation situations, a dynamic time slot allocation scheme of CQM protocol is proposed, based on traffic prediction.

TABLE 3: The parameter values of the BSA algorithm.

Parameters	C	S	a1	a2	FL	FQ	T	N
Values	1.5	1.5	1	1	FL ∈ [0.5, 0.9]	3	50	100

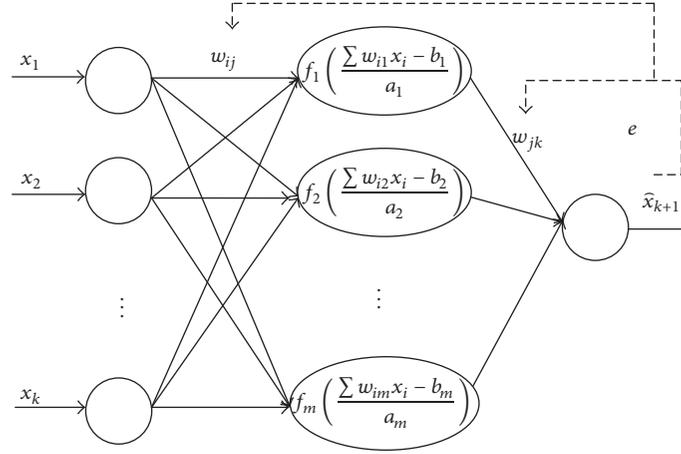


FIGURE 8: The prediction model of WNN.

In traffic prediction slot, each node predicts the real time arrival traffic  $W_{tra}$  by wavelet neural network. And then, in allocation slot, the next channel slot is allocated according to the predicted traffic of network.

**5.1. The Prediction Model of Wavelet Neural Network.** By contrast with the traditional neural network, WNN shows a higher accuracy, faster convergence, and better fault tolerance to complex nonlinear, uncertain, and unknown system. Wavelet neural network takes the topological structure of BP neural network as the foundation and selects the Morlet wavelet basis function as the transfer function of the hidden layer [27, 28]. As a feed-forward network, BP neural network includes the forward propagation of signal and the backpropagation of error. While the network is in the training and learning process, the weight is continuously adjusted to obtain the minimum total error which is relevant to the appropriate output of the network. The  $k - m - 1$  prediction model of WNN is shown in Figure 8.

As is shown in Figure 8, where  $x_i, i = 1, 2, \dots, k$ , is the input parameter,  $\hat{x}_{k+1}$  is the predicted output value in  $k + 1$  moment,  $w_{ij}$  is the layer weight from input layer to hidden layer, and  $w_{jk}$  is the layer weight from hidden layer to output layer, where  $m$  denotes the node number in hidden layer;  $a$  is the expansion parameter of the wavelet function;  $b$  is the translation parameter of the wavelet function. The Morlet wavelet basis function can extract the amplitude and phase information of the analyzed signal. Here the Morlet wavelet basis function as the activation function is shown as follows:

$$f(x) = \cos(1.75x) \exp\left(-\frac{x^2}{2}\right). \quad (43)$$

The wavelet neural network adopts the gradient descent algorithm [29] to correct the connection weight. The error

function  $E = (x_{k+1} - \hat{x}_{k+1})$  is used to correct the connection weight, so as to minimize the network error.

**5.2. The Dynamic Time Slot Allocation Scheme.** To get the optimal performance in both unsaturation and saturation situation, the DCQM protocol is proposed. Based on CQM protocol and WNN, the traffic prediction slot and allocation slot are added in CQM protocol. Figure 10 is an example of DCQM operation under  $Z_6$  with two channels (numbered from 0 to 1). Nodes 0, 1, and  $N$ , with IDs 0, 1, and  $N$ , respectively, are within each other's transmission range. The protocol processing of DCQM protocol is shown in Figure 9.

As is shown in Figure 9, the traffic prediction slot and allocation slot are added in front of each channel slot. In the end of a channel slot, number of packets transmitted in channel slot and that in the former channel slots are sent to WNN prediction processing. In traffic prediction slot, the predicted traffic to be transmitted in next channel slot of each node can be obtained. And during the allocation slot, if the predicted traffic is 0, the next channel slot is turned to default slot. Otherwise, nodes will broadcast their predicted traffic in their current channel in a random slot. Each node can get the predicted traffic of nodes in same channel, which can denote the traffic of network. And then, the next channel slot of each node can be allocated, according to one judgment criteria.

According to analysis of Figure 9, the judgment criteria are the key to allocate the channel slot. To utilize the channel efficiently, the switching channel slot should be preferentially allocated to the nodes that having packets to be transmitted. However, when more and more nodes have packets to be transmitted, if all of them are allocated to switching channel slot, the problem of missing destination node exists. To avoid this problem, the probability  $P_{allo\_switch}$  that the switching channel slot is allocated to a node is introduced. And let

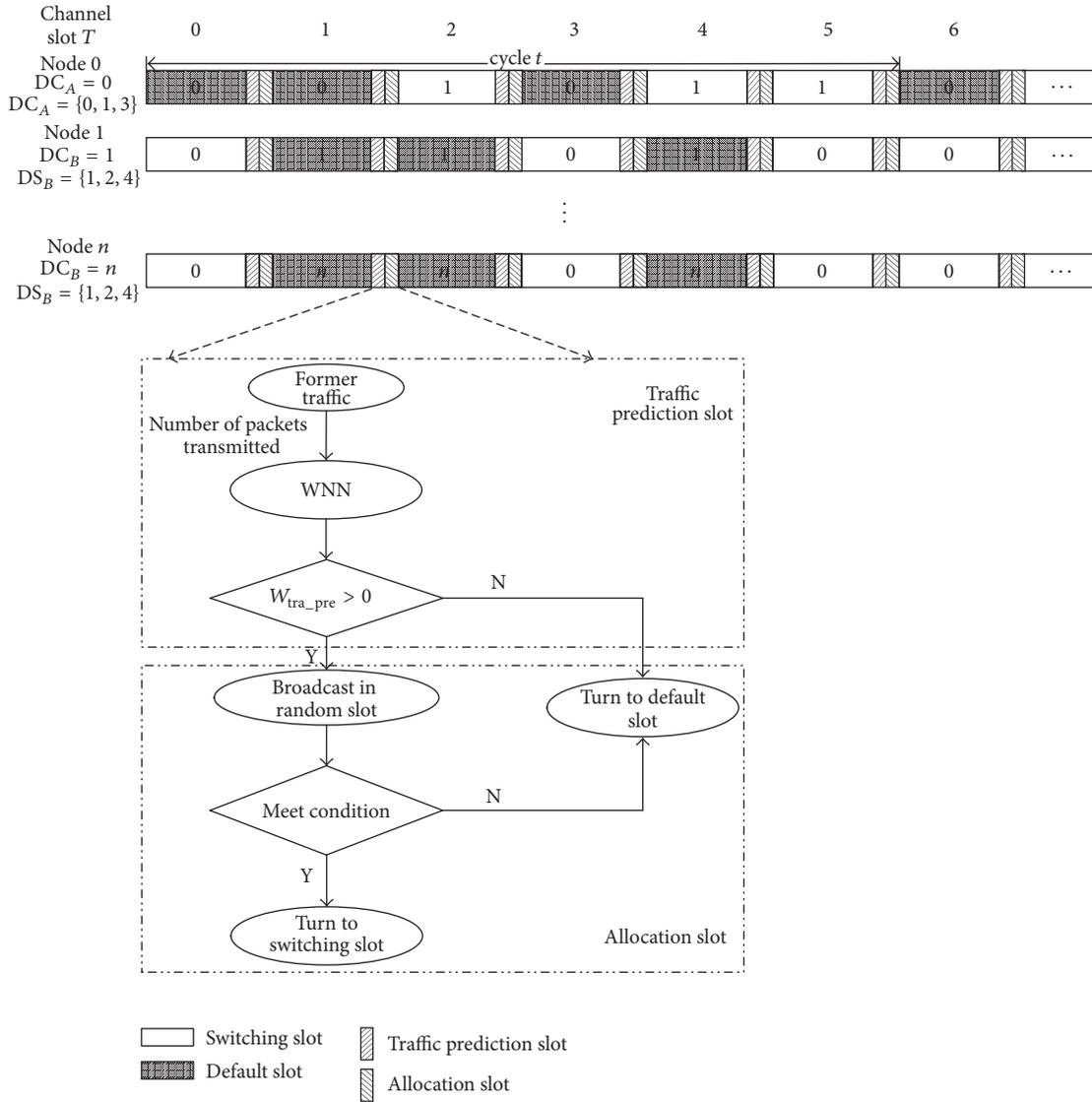


FIGURE 9: The protocol processing of DCQM protocol with two channels.

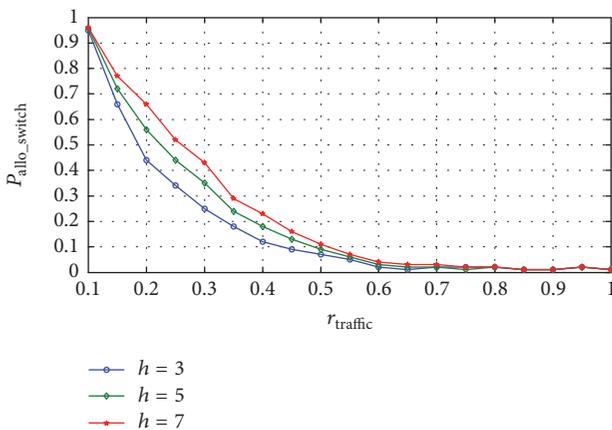


FIGURE 10: The relationship between  $P_{\text{allo\_switch}}$  and  $r_{\text{traffic}}$ .

$r_{\text{traffic}}$  denote the ratio of busy node number to network node number. And  $P_{\text{allo\_switch}}$  should meet the conditions as follows:

$$P_{\text{allo\_switch}} \begin{cases} \propto \frac{1}{r_{\text{traffic}}}, & 0 < r_{\text{traffic}} < 1 \\ = 1, & r_{\text{traffic}} \text{ is small enough} \\ = 0, & r_{\text{traffic}} \text{ is big enough.} \end{cases} \quad (44)$$

As is shown in (44),  $P_{\text{allo\_switch}}$  is inversely proportional to  $r_{\text{traffic}}$ . And  $r_{\text{traffic}}$  is obtained through broadcast in allocation slot. After traffic prediction slot, each node obtained their predicted packet number by WNN algorithm. During allocation slot, the nodes that having packets to be transmitted broadcast their predicted packet number in random slot. In this way, each node can obtain the approximate  $r_{\text{traffic}}$  of

network. And then, each node will turn to switching channel slot according to corresponding  $P_{\text{allo\_switch}}$ .

To obtain the optimal  $P_{\text{allo\_switch}}$  that meets the condition in (44), lots of Monte Carlo simulations are adopted, with  $h = 3, 5, 7$ , and 60 nodes considered. For one unsaturation situation ( $r_{\text{traffic}} \in [0.1, 1]$ ), lots of value of  $P_{\text{allo\_switch}} \in [0.01, 1]$  are considered. The optimal  $P_{\text{allo\_switch}}$  is obtained when the throughput of network reaches its maximum. The relationship between  $P_{\text{allo\_switch}}$  and  $r_{\text{traffic}}$  is revealed in Figure 10.

As is shown in Figure 10,  $P_{\text{allo\_switch}}$  is inversely proportional to  $r_{\text{traffic}}$ , which meets the condition in (44). And for the same  $r_{\text{traffic}}$ ,  $P_{\text{allo\_switch}}$  increases with the increase of channel number. With more channel numbers, the probability that nodes meet their destination nodes becomes bigger. And the utilization rate of frequency increases with the increase of channel number.

### 6. Performance Evaluation

To verify our analytical results, we compared analytic results with simulation results obtained from Qualnet simulator. In addition, the performance of CQM and DCQM protocols is also compared through Qualnet simulator. Qualnet simulator is the only parallel and message level network simulation tool, developed by Scalable Networks Technologies. It has the faster running speed, higher accuracy, and better extensibility and is suitable for development and simulation of ad hoc network and wireless sensor networks.

In our simulations, nodes are uniformly placed in an area of  $170 \text{ m} \times 170 \text{ m}$ . The transmission range of a node is 250 m. In this way, all nodes can stay in the transmission range of others. A node keeps a separate FIFO queue for each of its neighbors. Each node may act as a sender where the destination is chosen from its one-hop neighbors. The packet rate is selected in [2, 6]. And the channel slot is set to 100 ms. To get the performance of different channel number, every node is equipped with several interfaces. The interface can be assumed to be different channels, and the CQM channel-hopping strategy is used in our simulations. The number of interface is set to 3, 5, and 7. The system parameters are the same as the numerical analysis in Section 3.

Firstly, we will verify the saturation bound of analysis results. Figure 11 shows results comparison between numerical analysis and simulation when  $\lambda$  is set to 2, 3, 4, 5, 6, 7, and 8.

As is shown in Figure 11, the numerical analysis results can be verified by simulating. In addition, the more nodes can be accommodated with the increase of channel number. It can be verified that the multichannel system can perform better than the single system in high diversity node situation.

And then, we present the aggregate throughput between the CQM and proposed DCQM protocol as shown in Figure 12. The simulation parameters of DCQM protocol are the same as that of the CQM protocol. In addition, for DCQM,  $T_{\text{prediction}}$  and  $T_{\text{allocation}}$  denote the time in the traffic prediction stage and allocation stage, respectively. In this simulation,  $T_{\text{prediction}}$  is set to 1ms, and the  $T_{\text{allocation}}$  is set to 5 ms. Each traffic flow in the network uses the constant bit

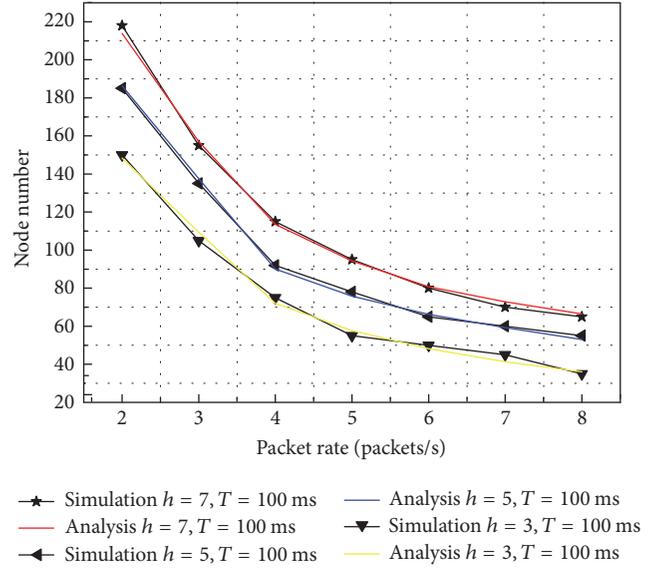


FIGURE 11: The comparison between analysis results and simulation results with different channel number.

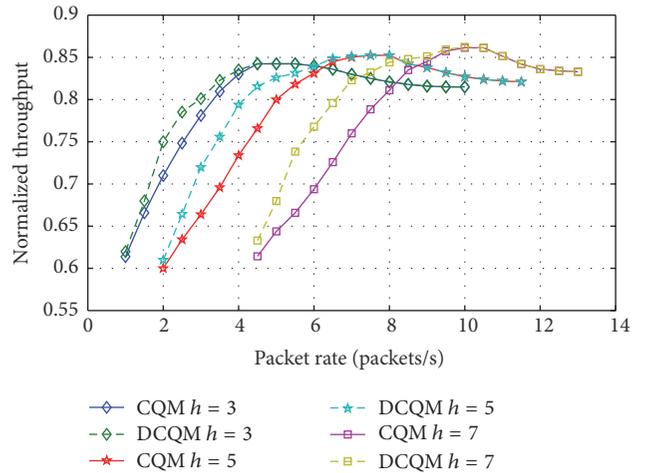


FIGURE 12: The performance comparison between CQM and DCQM protocol.

rate (CBR) traffic model, the packet size is 1024 bytes, and the packet rate is set to 8.

As is shown in Figure 12, for CQM and DCQM protocols, with different channel number, the normalized throughput increases with the increase of packet rate. And then, it reaches its maximum value when node number reaches one certain value. Finally, it decreases with the increase of node number gradually. In addition, in unsaturation situation, the normalized throughput of DCQM is bigger than that of CQM protocol. This is because the utilization of frequency increases due to dynamic channel slot allocation. And in saturation situation, the performance of CQM protocol is slightly better than DCQM protocol. For DCQM protocol, in saturation situation,  $P_{\text{allo\_switch}}$  is almost 0, and the channel slot allocation is the same as CQM protocol. However, the additional overhead is needed in DCQM protocol because of

the additional traffic prediction slot  $T_{\text{prediction}}$  and allocation slot  $T_{\text{allocation}}$ .

## 7. Conclusions

IEEE 802.11 DCF suffers from many collisions in high diversity node situation. Multichannel MAC protocol can help to share the traffic loads among different channels. And the CQM protocol performed better among MMAC protocols with only one transceiver. However, the performance of CQM protocol is not analyzed theoretically. And the performance is not optimal for the fixed channel slot allocation scheme in unsaturation situation.

In this paper, a Markov chain model is proposed to analyze the performance of CQM protocol theoretically. The throughput and average packet transmission delay of CQM system are revealed too. And the saturation bound of CQM protocol is obtained based on BSA algorithm. In addition, to obtain the optimal performance of CQM protocol in unsaturation situation, a traffic prediction based dynamic channel slot allocation scheme of CQM protocol is proposed based on WNN prediction model. In the end, the performance theoretical analysis of CQM protocol is verified by simulating on Qualnet platform. And the performance comparison of CQM and DCQM protocol is simulated too. The results show that the simulation results and analytical results match very well, and the DCQM protocol performs better than CQM protocol in unsaturation situation.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

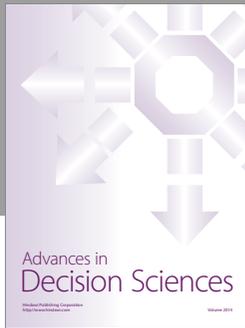
## Acknowledgments

Xing HU wishes to thank Lin-hua MA for their helpful discussions on information theoretic aspects of the subject. This work has been supported by the Nature Science Foundation of China (61072102) and the State Key Laboratory of Xidian University (ISN15-13).

## References

- [1] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, 2000.
- [2] C.-M. Chao and H.-C. Tsai, "A channel-hopping multichannel MAC protocol for mobile ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 9, pp. 4464–4475, 2014.
- [3] K. H. Almotairi and X. Shen, "Multichannel medium access control for ad hoc wireless networks," *Wireless Communications and Mobile Computing*, vol. 13, no. 11, pp. 1047–1059, 2013.
- [4] D. N. M. Dang and C. S. Hong, "H-MMAC: A hybrid multi-channel MAC protocol for wireless ad hoc networks," in *Proceedings of the 2012 IEEE International Conference on Communications, ICC 2012*, pp. 6489–6493, can, June 2012.
- [5] J. Shi, T. Salonidis, and E. W. Knightly, "Starvation mitigation through multi-channel coordination in CSMA Multi-hop wireless networks," in *Proceedings of the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing, (MOBIHOC '06)*, pp. 214–225, ACM, India, May 2006.
- [6] M. Maiya and B. Hamdaoui, "An improved IEEE 802.11 MAC protocol for wireless ad-hoc networks with multi-channel access capabilities," in *Proceedings of the 2011 International Conference on High Performance Computing and Simulation, HPCS 2011*, pp. 162–168, tur, July 2011.
- [7] H. Lei, Z. Ren, C. Gao, and Y. Guo, "A new multi-channel MAC protocol for 802.11-based wireless mesh networks," in *Proceedings of the 2012 International Conference on Computer Science and Electronics Engineering, ICCSEE 2012*, pp. 27–31, chn, March 2012.
- [8] O. D. Incel, L. Van Hoesel, P. Jansen, and P. Havinga, "MC-LMAC: A multi-channel MAC protocol for wireless sensor networks," *Ad Hoc Networks*, vol. 9, no. 1, pp. 73–94, 2011.
- [9] T. Luo, M. Motani, and V. Srinivasan, "Cooperative asynchronous multichannel MAC: Design, analysis, and implementation," *IEEE Transactions on Mobile Computing*, vol. 8, no. 3, pp. 338–352, 2009.
- [10] J. So and N. Vaidya, "Multi-channel MAC for Ad Hoc networks: Handling multi-channel hidden terminals using a single transceiver," in *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing*, pp. 222–223, ACM, India, 2004.
- [11] J. Wang, Y. Fang, and D. O. Wu, "A power-saving multi-radio multi-channel MAC protocol for wireless local area networks," in *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM '06)*, pp. 1–12, IEEE, Barcelona, Spain, April 2006.
- [12] P.-J. Wu and C.-N. Lee, "On-demand connection-oriented multi-channel MAC protocol for ad-hoc network," in *Proceedings of the 2006 3rd Annual IEEE Communications Society on Sensor and Ad hoc Communications and Networks, Secon 2006*, pp. 621–625, usa, September 2006.
- [13] S.-L. Wu, Y.-C. Tseng, C.-Y. Lin, and J.-P. Sheu, "A multi-channel MAC protocol with power control for multi-hop mobile ad hoc networks," *Computer Journal*, vol. 45, no. 1, pp. 101–110, 2002.
- [14] S.-L. Wu, C.-Y. Lin, Y.-C. Tseng, and J.-P. Sheu, "A new multi-channel MAC protocol with on-demand channel assignment for multi-hop mobile ad hoc network," *Parallel Architectures, Algorithms and Networks*, pp. 232–237, 2000.
- [15] Z. Xiangquan, G. Lijia, and G. Wei, "A load-balanced MAC protocol for multi-channel ad-hoc networks," in *Proceedings of the ITST 2006 - 2006 6th International Conference on ITS Telecommunications, Proceedings*, pp. 642–645, chn, June 2006.
- [16] J.-H. Kim and S.-J. Yoo, "TMCMP: TDMA based multi-channel MAC protocol for improving channel efficiency in wireless Ad hoc networks," in *Proceedings of the 2009 IEEE 9th Malaysia International Conference on Communications with a Special Workshop on Digital TV Contents, MICC 2009*, pp. 429–434, mys, December 2009.
- [17] P. Kysanur and N. H. Vaidya, "Routing and link-layer protocols for multi-channel multi-interface ad hoc wireless networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 10, no. 1, pp. 31–43, 2006.
- [18] J. S. Pathmasuntharam, A. Das, and A. K. Gupta, "Primary channel assignment based MAC (PCAM)-A multi-channel MAC protocol for Multi-Hop wireless networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '04)*, vol. 2, pp. 1110–1115, March 2004.
- [19] H.-S. W. So, G. Nguyen, and J. Walrand, "Practical synchronization techniques for multichannel MAC," in *Proceedings of*

- the 12Th Annual International Conference on Mobile Computing And Networking*, pp. 134–145, ACM, India, 2006.
- [20] L. Tang, Y. Sun, O. Gurewitz, and D. B. Johnson, “EM-MAC: a dynamic multichannel energy-efficient MAC protocol for wireless sensor networks,” in *Proceedings of the 12th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc’11)*, ACM, Las Vegas, Nev, USA, May 2011.
- [21] P. Bahl, R. Chandra, and J. Dunagan, “SSCH: Slotted seeded channel hopping for capacity improvement in IEEE 802.11 Ad-Hoc wireless networks,” in *Proceedings of the MobiCom 2004 - Proceedings of the Tenth Annual International Conference on Mobile Computing and Networking*, pp. 216–230, India, October 2004.
- [22] 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 Transactions on Mobile Computing*, vol. 5, no. 5, pp. 560–570, 2006.
- [23] J.-R. Jiang, Y.-C. Tseng, C.-S. Hsu, and T.-H. Lai, “Quorum-based asynchronous power-saving protocols for IEEE 802.11 ad hoc networks,” *Mobile Networks and Applications*, vol. 10, no. 1-2, pp. 169–181, 2005.
- [24] P. Chatzimisios, A. C. Boucouvalas, and V. Vitsas, “Performance analysis of the IEEE 802.11 MAC protocol for wireless LANs,” *International Journal of Communication Systems*, vol. 18, no. 6, pp. 545–569, 2005.
- [25] IEEE Computer Society LAN MAN Standards Committee. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications[J], 1997.
- [26] X. B. Meng, X. Z. Gao, L. H. Lu et al., “A new bio-inspired optimisation algorithm: Bird Swarm Algorithm,” *Journal of Experimental Theoretical Artificial Intelligence*, pp. 1–15, 2015.
- [27] H. Yang, W. Zhao, W. Chen, and X. Chen, “The study of prediction model based on morlet wavelet neural network,” *Applied Mechanics and Materials*, vol. 121-126, pp. 4847–4851, 2012.
- [28] Y.-S. Lu, B.-X. Wu, and S.-F. Lien, “An improved sliding-mode repetitive learning control scheme using wavelet transform,” *Asian Journal of Control*, vol. 14, no. 4, pp. 991–1001, 2012.
- [29] Y. Y. Yan and B. L. Guo, “Application of wavelet neural network (WNN) and gradient descent method (GDM) in natural image denoising,” *Journal of Computational Information Systems*, vol. 2, no. 2, pp. 625–631, 2006.



# Hindawi

Submit your manuscripts at  
<https://www.hindawi.com>

