Neural Comput & Applic (2014) 25:1845–1851
DOI 10.1007/s00521-014-1674-1

ORIGINAL ARTICLE

An adaptive backoff algorithm for multi-channel CSMA in wireless sensor networks

Yantao Li · Gang Zhou · Nan Zheng · Liang Hong

Received: 18 January 2014 / Accepted: 14 July 2014 / Published online: 2 August 2014

Abstract Multiple channels have been widely used in wireless sensor networks (WSNs) for the improvement of network performance. Since the existing backoff algorithms proposed for single-channel MAC protocols are not suitable for multi-channel carrier sense multiple access (CSMA), we propose an ADaptive Backoff Algorithm (ADBA) for multi-channel CSMA in WSNs, which is able to improve energy efficiency, throughput, and fairness of random channel accesses. A novel feature of ADBA is the adaptability to traffic load, where every node tunes its backoff interval based on the estimation of real-time traffic load in WSNs. A near-optimal backoff interval can be generated using the number of competing nodes on one channel which can be estimated by the channel traffic load. Theoretical analysis indicates that ADBA can generate near-optimal backoff intervals that can maximize energy efficiency and throughput and improve fairness of random channel accesses, compared with other backoff schemes.

Keywords Algorithm · Adaptive · Multiple channels · Near-optimal · Wireless sensor networks

Y. Li (✉)
College of Computer and Information Sciences, Southwest University, Chongqing 400715, China
e-mail: yantaoli@foxmail.com; liyantao@live.com; yantaoli@swu.edu.cn

G. Zhou · N. Zheng
Department of Computer Science, College of William and Mary, Williamsburg, VA 23187, USA

L. Hong
School of Computer, Wuhan University, Wuhan 430072, Hubei, China

1 Introduction

The proliferation of hardware platforms and protocols for multi-channel communications in wireless sensor networks (WSNs) provides more reliable and timely communication services for a wide range of applications. In real deployment, it is possible for multiple WSNs running in adjacent physical locations. Thus, more sophisticated MAC protocols should be designed to coordinate channel switching and transmissions among nodes in multi-channel WSNs [14, 17, 19]. Multi-channel MAC protocols in WSNs have been proved that they are able to improve the network throughput and reduce the probability of radio frequency (RF) conflicts [3, 21, 22], among which carrier sense multiple access (CSMA) protocol is one of the most widely used channel-access approaches in terms of its simplicity, flexibility, and robustness [11]. More specifically, *carrier sense* means that a sender determines whether another transmission is in progress before initiating a transmission using feedback from a receiver. That is to say, the sender attempts to detect the presence of a carrier wave from another station before trying to send. If a carrier is sensed, the station will wait for the transmission in progress until it is done before initiating its own transmission. In other words, we can say that CSMA is based on the principle of "sense before transmit" or "listen before talk." On the other hand, *multiple access* means that multiple stations send and receive on the medium. Transmissions from one node are generally received by all other stations connected to the medium. For CSMA, backoff happens whenever a sending host encounters a collision. As we know, a collision occurs when at least two sending hosts choose the same channel to transmit packets. If these hosts continue to retransmit packets right after a collision occurs, then they will suffer consecutive collisions. To avoid this, the hosts must choose a random *backoff interval* (i.e., waiting time) within an acceptable range typically called a *contention window*. Thus, a backoff algorithm is designed to dynamically control the backoff interval, which will increase backoff interval for the next transmission when suffering a collision and decrease it after a successful transmission. The underlying rationale is to regard a transmission collision as a signal of increasing backoff interval for retransmissions.

Although the backoff algorithm is of vital importance in multi-channel communications, to the best of our knowledge, little attention has been paid to the backoff algorithm for multi-channel CSMA in WSNs. In multi-channel CSMA, the total traffic load dynamically spreads over all channels and the traffic load varies on a particular channel continuously. Such a characteristic brings challenges to the design of a backoff algorithm for multi-channel CSMA, since contention level on a channel varies continuously with the real-time traffic load and the backoff interval should be tuned accordingly. Unfortunately, the existing backoff algorithms are almost designed for single-channel MAC protocols and do not take the dynamic traffic load on every channel into account. The collision resolution strategy binary exponential backoff (BEB) [4, 8, 10] is initially used in the MAC protocols of both Ethernet [16] and wireless LAN [8]. Compared with BEB strategy, the multiplicative increase linear decrease (MILD) backoff scheme in MACAW protocol [5, 13], however, greatly improves the fairness of transmitters. In the sensing backoff algorithm (SBA) [6], every node modifies its backoff interval according to the results of sensed channel activities. However, BEB, MILD, and SBA do not consider the dynamic channel traffic load, and they cannot generate optimal backoff intervals for multi-channel CSMA. We are different in that we consider dynamic channel traffic load.

In this paper, we propose an ADaptive Backoff Algorithm (ADBA) for multi-channel CSMA in WSNs, which is able to dynamically choose the backoff interval that adapts to the current contention level on the channel (i.e., the number of nodes that compete for the same channel) using a system control loop. The contention level is estimated by the real-time traffic load that can be obtained using a middleware self-adaptive spectrum (SAS) [18, 23], which is a SAS management middleware for WSNs that allows an existing single-frequency MAC protocol to be self-adaptive and take advantage of multiple frequencies for parallel communication. ADBA takes the dynamic traffic load distribution into account and is especially designed to optimize the network performance in multi-channel communications. The main contributions of this paper can be summarized as follows:

- To the best of our knowledge, we are among the first to present the ADBA that is especially designed for multi-channel CSMA in WSNs, which is able to adaptively choose the backoff interval.
- We formulate the optimal backoff interval that is a function of the number of contenders based on the estimation of the real-time channel traffic load.
- We conduct a strict theoretical analysis on the performance of ADBA and indicate that ADBA can achieve optimal energy efficiency, throughput, and fairness of random channel accesses for multi-channel CSMA in WSNs.

The remainder of this paper is organized as follows: Sect. 2 discusses related work for the existing backoff algorithms. In Sect. 3, we present the design of ADBA algorithm in detail. Then, we conduct a theoretical analysis on the performance of ADBA algorithm in Sect. 4 and present the conclusions in Sect. 5.

2 Related work

The BEB is a collision resolution strategy that is initially used in the MAC protocols of both Ethernet [16] and wireless LAN [8]. BEB exponentially sets backoff intervals according to the number of collisions in data transmissions. As the number of packet retransmissions increases, the time for delay increases. Various studies [4, 10, 15] show that BEB is unstable for an infinite-node model or a finite-node model, if the packet arriving rate is high enough. Moreover, BEB strategy suffers fairness issue in some scenarios [7]. Compared with BEB strategy, the MILD backoff scheme in MACAW protocol [5, 13], however, greatly improves the fairness of transmitters. In the MACAW protocol, the backoff interval copy mechanism is implemented in each of the nodes, which copies the backoff intervals of the overheard successful transmitters. The MILD scheme can be summarized as follows:

$$\begin{aligned} &x \leftarrow \min(1.5x, B_{\max}), && \text{upon collision} \\ &x \leftarrow x_{\text{packet}}, && \text{upon successful overhearing} \\ &x \leftarrow \max(x-1, B_{\min}), && \text{upon successful transmission} \end{aligned}$$

where x is the backoff interval value, x_{packet} is the backoff interval value included in the overheard packet, and $B_{\min}$ and $B_{\max}$ are the minimum and maximum backoff interval values predetermined based on the possible number of active nodes and the network traffic load.

With the implemented copy mechanism, the fairness performance of the MILD scheme is greatly improved. However, the backoff interval included in the packets increases the length of the overhead. Thus, the probability of packet collisions is decreased. In addition, copy mechanism also results in the migration of the backoff intervals that would decrease the throughput of the channel.

To address the channel domination problem that caused by BEB and the backoff interval migration issue that caused by MILD, the SBA is proposed [6]. In SBA, every node modifies its backoff interval according to the results of sensed channel activities. The SBA can be summarized by the following set of equations:

$$\begin{aligned} &x \leftarrow \min(1.2x, B_{\max}), && \text{upon collision} \\ &x \leftarrow \max(x - 0.8\gamma, B_{\min}), && \text{upon sensing successful packet} \\ &x \leftarrow \max(0.93x, B_{\min}), && \text{upon successful transmission} \end{aligned}$$

where γ is defined as the time for transmission of a packet.

As we mentioned above, in multi-channel CSMA, the underlying channel varies dynamically during the backoff period. As a result, the traffic load also varies with the variation of the channel. Since single-channel backoff algorithms such as BEB [4, 8, 10], MILD [5, 13], and SBA [6] do not consider the dynamic channel traffic load, they cannot generate optimal backoff intervals for multi-channel CSMA.

In [9], the authors propose the linear/MILD (LMILD) backoff algorithm, which is based on the IEEE 802.11 Distributed Coordination Function. The authors in [20] propose a multichain backoff (MCB) algorithm that allows stations to adapt to different congestion levels using more than one backoff chain together with collision events caused by stations themselves as well as other stations as indications for choosing the next backoff chain.

In order to obtain an optimal backoff interval for multi-channel CSMA, a backoff algorithm should take the time-varying traffic load into account, that is, adaptively tune the backoff interval according to the real-time traffic load. Then, another issue is how to obtain the time-varying traffic load information. We exploit a recently developed middleware for CSMA self-adaptive spectrum (SAS) that is able to perform dynamic estimation on traffic load in multi-channel WSNs [18, 23]. Based on SAS, we propose a new backoff algorithm ADBA that tunes the backoff interval dynamically according to the current channel traffic load.

3 ADBA algorithm

In this section, we first formulate the problem of generating optimal backoff intervals for multi-channel CSMA and then present the design of ADBA algorithm in detail using the control theory as a scientific underpinning.

3.1 Problem formulation

We consider the problem of deciding backoff intervals for CSMA on a shared channel i in multi-channel WSNs. The backoff algorithm for multi-channel CSMA operates as follows:

- A new packet arrives at a node that is currently transmitting packets over channel i. Before the transmission of a packet, the node generates a random backoff waiting time according to the uniform distribution between 0 and its backoff interval B_i. All the nodes in the WSN have the same value of B_i.
- At the end of the random backoff waiting time, the packet will be transmitted.
- If the packet transmission is unsuccessful, a new random backoff waiting time will be generated and the packet will be retransmitted once the new backoff waiting time expires.

Suppose there are K time slots on channel i for CSMA competition. We say that there is a collision on slot r if at least two nodes choose the slot r. A node wins slot r if and only if it is the only node choosing slot r. Slot r is idle if no node chooses r, otherwise slot r is busy if either a node wins r, or there is a collision on r. We define traffic load L_i over K time slots on channel i. as:

$$L_i = \frac{N_{\text{busy}}}{K} = 1 - \frac{N_{\text{idle}}}{K} \tag{1}$$

where N_{busy} denotes the number of busy slots and N_{idle} denotes the number of idle slots.

In the multi-channel communication of WSNs, the number of sensor nodes that compete for a channel i may be time-varying; thus it affects the number of busy slots. For a sensor node that transmits packets over channel i, our problem can be refined to decide an optimal backoff interval B_i^* that should be a function of current traffic load L_i, which could optimize network performance such as channel throughput, fairness of random channel accesses, and delay.

We think of the operation of ADBA as a control loop, as shown in Fig. 1. As illustrated in this figure, we first obtain the real-time traffic load information that is estimated by implementation of the middleware SAS in Block A, which is a SAS management middleware for WSNs that allows an existing single-frequency MAC protocol to be self-adaptive and take advantage of multiple frequencies for parallel communication. Then, the number of active nodes (called *contender*) that compete for current channel is estimated in Block B. The new optimal backoff interval is then decided based on the estimated number of contenders in Block C.

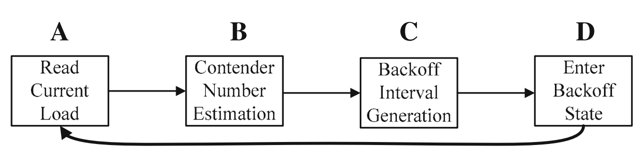

Fig. 1 ADBA control loop

Finally, the node enters its backoff state and waits for a random time chosen between 0 and the new backoff interval in Block D. Upon the expiration of backoff waiting time, the node retransmits the packet; in the case of a collision again, the same loop will be executed.

Since we can already estimate current traffic load by SAS, we will propose a novel method to estimate the number of contenders on a channel in the following. Based on that, we will generate an optimal backoff interval for multi-channel CSMA.

3.2 Contender number estimation

Consider a contention window (backoff interval) consisting of K time slots. Suppose there are N contenders for channel i. Each of them randomly chooses one slot from the K time slots of the contention window. A slot has either of the two states, busy or idle. The states of slot r are defined as follows:

$$\begin{cases} \text{idle,} & \text{if no contender chooses slot } r \\ \text{busy} & \text{if only one contender chooses slot } r \\ \text{busy} & \text{if at least two contenders choose slot } r \end{cases}$$

There are two situations when slot r is busy. If slot r is chosen by more than one contender, a collision occurs, so none of the contenders can win slot r. Otherwise, if slot r is chosen by exactly one contender, the contender wins slot r.

As mentioned above, we need to estimate the number of contenders on a specific channel. The first step is to estimate the number of idle slots. After all contenders randomly choose their slots, the probability for one particular slot r being idle is:

$$P_{\text{idle}}(r) = \left(1 - \frac{1}{K}\right)^N \approx e^{\frac{-N}{K}} \tag{2}$$

where the last approximation holds when N is sufficiently large. Thus, the expected number of idle slot $E(N_{\text{idle}})$ can be calculated as:

$$E(N_{\text{idle}}) \approx Ke^{\frac{-N}{K}} \tag{3}$$

According to Eq. (1), traffic load on channel i can be defined as a function of the number of idle slots N_{idle}. Substituting Eq. (3) into Eq. (1), we have the following equation:

$$L_i = 1 - \frac{E(N_{\text{idle}})}{K} \approx 1 - e^{\frac{-N}{K}} \tag{4}$$

From Eq. (4), the estimated number of contenders on channel i $\hat{N}_i$ can be expressed as:

$$\hat{N}_i = \lceil -K \ln(1 - L_i) \rceil \tag{5}$$

Equation (5) is used in Block B in Fig. 1 to estimate the number of contenders on a particular channel based on the current channel traffic load.

3.3 Generation of optimal backoff interval

The relationship between the optimal backoff interval and the number of active nodes (i.e., the number of contenders N) has been studied in [6]. For a particular shared channel, the optimal backoff interval can be calculated as:

$$B_{\text{opt}}(N) = 4N\gamma \tag{6}$$

where N is the number of contenders that compete for the current channel, and γ is the time for the transmission of a packet. The backoff interval B_{opt} is optimal in the sense that it maximizes the channel throughput and improves the fairness of random channel accesses. In CSMA, the channel throughput describes the fraction of slots at which packets are successfully transmitted [12]. However, Haas and Deng [6] do not provide the mechanism that estimates the number of contenders on a particular channel for multi-channel MAC protocols. Thus, it cannot give an optimal backoff interval as a function of the number of contenders for multi-channel CSMA.

Utilizing the result in Eq. (6), the optimal backoff interval B^* on channel i can be computed as:

$$B^* = B_{\text{opt}}(\hat{N}_i) = 4\hat{N}_i\gamma \tag{7}$$

where $\hat{N}_i$ is the estimated number of contenders on channel i using Eq. (5).

When a collision occurs, each colliding node first generates the optimal backoff interval B^* using Eq. (7), then chooses random number of time slots between 0 and B^*, and enters its backoff state for the backoff waiting time (Block D). Once the backoff waiting time expires, the colliding node attempts to retransmit its packet. If the node wins a certain slot, it exits the control loop in Fig. 1 and, meanwhile, successfully transmits the packet. Otherwise, it continues in Blocks B and C of the control loop to generate an up-to-date optimal backoff interval.

3.4 The ADBA algorithm

In Fig. 1, ADBA is perceived as a control loop that controls the backoff for multi-channel CSMA. The ADBA algorithm is illustrated in Algorithm 1 in detail. In Algorithm 1, real-time traffic load L_i on channel i is first estimated using middleware SAS. Then, based on L_i, the number of contenders on channel i is estimated. Next, the optimal backoff interval is generated with the number of current contenders. Finally, each contender on channel i chooses random numbers of time slots between 0 and optimal backoff interval.

Algorithm 1: The algorithm of ADBA

```
Input: Traffic load L_i on channel i(C_i)
Output: The backoff waiting time W_ij for node j on C_i
for each channel C_i do
    L_i = SAS(C_i)
    N̂_i = ⌈−K ln(1 − L_i)⌉
    for each active node on C_i do
        B* = 4 N̂_i γ
    end
end
for active node j on C_i do
    W_ij = GetRand(0, B*)
    Node j backs off for W_ij
end
```

For each contender on channel i, it computes the optimal backoff interval using ADBA. As a result, the contenders on the same channel share the same optimal backoff interval, which guarantees the fairness of the random channel accesses. The adaptability to ADBA is achieved in the sense that the output (backoff waiting time) is controlled by the input (traffic load) in the control system.

4 Theoretical analysis

In this section, we theoretically analyze several performance metrics of the proposed backoff algorithm in terms of channel throughput, energy consumption, fairness of random channel accesses, and packet delay.

4.1 Analysis of channel throughput

The optimal backoff interval B^* can be proved that it maximizes the channel throughput using the results in [6]. That is, we can achieve the maximum channel throughput if we maintain B as close as possible to the optimal backoff interval B^*.

4.2 Analysis of energy consumption

According to [2], the average energy consumed by a sensor node during transmission can be calculated as:

$$\begin{aligned} &E[\text{Energy}_{\text{transmission}}] \\ &= E[\text{Energy}_{\text{succ}}] \times \text{Prob}_{\text{succ}} + \text{E}[\text{Energy}_{\text{coll}}] \times \text{Prob}_{\text{coll}} \\ &= \text{PTX} \times \gamma \times \text{Prob}_{\text{succ}} + (\text{PTX} \times \gamma + \text{PRX} \times \gamma) \times \text{Prob}_{\text{coll}} \end{aligned} \tag{8}$$

where $E[\text{Energy}_{\text{succ}}]$ is the average energy consumed by a successful transmission; $\text{Prob}_{\text{succ}}$ is the probability of a successful transmission in a transmission attempt; $E[\text{Energy}_{\text{coll}}]$ is the average energy consumed when a collision occurs; $\text{Prob}_{\text{coll}}$ is the probability of a collision; PTX is the energy consumption of transmission during every time unit; PRX is the energy consumption during receiving period; and γ is the transmission time for a packet.

According to [6], we obtain $\text{Prob}_{\text{succ}}$ and $\text{Prob}_{\text{coll}}$:

$$P_{\text{succ}} = \left(1 - \frac{2\gamma}{B}\right)^{N-1} \tag{9}$$

$$P_{\text{coll}} = 1 - \left(1 - \frac{2\gamma}{B}\right)^{N-1} \tag{10}$$

Substituting Eqs. (9) and (10) into Eq. (8), the average energy consumption can be calculated as:

$$\begin{aligned} E[\text{Energy}_{\text{transmission}}] = \text{PTX} \times \gamma + \text{PRX} \times \gamma \\ \times \left[1 - \left(1 - \frac{2\gamma}{B}\right)^{N-1}\right] \end{aligned} \tag{11}$$

As we can see from Eq. (11), the average energy consumption is a function of the number of contenders N and backoff interval B.

The total energy consumption of a sensor node for a packet transmission attempt consists of the energy consumed for transmitting and the energy consumed in the idle time. Since the average idle time $E[\text{Idle}]$ in a transmission attempt is $\frac{B}{2N}$ [6], the energy consumed in idle time can be calculated as:

$$E[\text{Energy}_{\text{idle}}] = \text{PRX} \times \text{E}[\text{Idle}] = \text{PRX} \times \frac{\text{B}}{2\text{N}} \tag{12}$$

The energy efficiency is the ratio between the energy used in a successful transmission and the total energy consumption for a transmission attempt. Thus, the energy efficiency ρ_{energy} can be expressed as:

$$\begin{aligned} \rho_{\text{energy}} &= \frac{E[\text{Energy}_{\text{succ}}]}{E[\text{Energy}_{\text{transmission}}] + \text{E}[\text{Energy}_{\text{idle}}]} \\ &= \frac{\text{PTX} \times \gamma \times \text{Prob}_{\text{succ}}}{\text{PTX} \times \gamma \times \text{Prob}_{\text{succ}} + (\text{PTX} \times \gamma + \text{PRX} \times \gamma) \times \text{Prob}_{\text{coll}} + \text{PRX} \times \text{E}[\text{Idle}]} \\ &= \frac{\text{PTX} \times \gamma \times \left(1 - \frac{2\gamma}{\text{B}}\right)^{\hat{\text{N}}-1}}{\text{PTX} \times \gamma \times \left(1 - \frac{2\gamma}{\text{B}}\right)^{\hat{\text{N}}-1} + (\text{PTX} + \text{PRX}) \times \gamma \times \left[1 - \left(1 - \frac{2\gamma}{\text{B}}\right)^{\hat{\text{N}}-1}\right] + \text{PRX} \times \frac{\text{B}}{2\text{N}}} \end{aligned} \tag{13}$$

Assume that PTX = PRX, then Eq. (13) can be written as:

$$\rho_{\text{energy}} = \frac{\gamma \times \left(1 - \frac{2\gamma}{B}\right)^{N-1}}{\gamma \times \left[2 - \left(1 - \frac{2\gamma}{B}\right)^{N-1}\right] + \frac{B}{2N}} \tag{14}$$

Based on the equation $\frac{\partial(\rho_{\text{energy}})}{\partial B} = 0$, it can be proved that when $B \approx 4N\gamma$, ρ_{energy} can be maximized. Therefore, energy efficiency and throughput maximization can be jointly achieved if and only if $B^{*} \approx 4\hat{N}\gamma$, which means ADBA can maximize energy efficiency and throughput.

4.3 Analysis of fairness

Since B^{*} is shared by all the contenders on the same channel, these contenders have the same probability to access the channel. That is, the fairness index (FI) [1] level on a specific channel is $\frac{1}{N}$, which represents the randomness of traffic generation.

4.4 Analysis of packet delay

The packet delay is referred to as the time used for packet propagation, packet transmission and retransmission, and acknowledgment transmission. Due to the local coverage of sensor networks, we neglect the propagation delay, and since acknowledgment packet is tiny, we neglect the transmission time of the acknowledgment packet as well. Thus, the average retransmission delay, that is the delay between two successfully consecutive transmissions of a given packet, can be calculated as [1]:

$$D = \left(\frac{G}{S} - 1\right)(\gamma + T_a + \bar{X}) + \gamma \tag{15}$$

where G is the packet arriving rate, S is the channel throughput, T_{a} is the transmission time of the acknowledgment packet, and $\bar{X}$ is the average backoff waiting time.

According to [6], we have $G = \frac{2N}{B}$ and $\bar{X} = \frac{B}{2}$, where N is the number of contenders and B is the backoff interval. By substituting the two, the average retransmission delay can be rewritten as:

$$D = \left(\frac{2N}{BS} - 1\right)\left(\gamma + \frac{B}{2}\right) + \gamma = \frac{2N\gamma}{BS} + \frac{N}{S} - \frac{B}{2} \tag{16}$$

where S can be expressed as a function of N and B in [6]. As a result, the average retransmission delay can be estimated by Eq. (9) if we obtain N and B.

5 Conclusions

In this paper, we propose ADBA, a new backoff algorithm that can generate an optimal backoff interval for each sensor node that uses CSMA in multi-channel communication. ADBA is able to adaptively estimate the number of contenders on a specific channel according to real-time traffic load. Based on the time-varying contender number, ADBA generates an optimal backoff interval. Compared with existing backoff algorithms, ADBA maximizes the channel throughput and reduces energy consumption by dynamically tuning the backoff interval according to current real-time traffic load. ADBA is valuable for multi-channel communications in resource-constrained WSNs.

Acknowledgments The work was supported by the National Natural Science Foundation of China (61173178, 61070246), the Fundamental Research Funds for the Central Universities (XDJK2013C116, XDJK2013B030), and the Doctoral Research Funds of Southwest University (SWU113019, SWU111047).

References

1. Bensaou B, Wang Y, Ko CC (2000) Fair medium access in 802.11 based wireless ad-hoc networks. In: IEEE/ACM MobiHoc Workshop, pp 99–106
2. Bruno R, Conti M, Gregori E (2002) Optimization of efficiency and energy consumption in p-persistent CSMA-based wireless LANs. IEEE Trans Mob Comput 1:10–31
3. Chen X, Han P, He QS, Tu SL, Chen ZL (2006) A multichannel MAC protocol for wireless sensor networks. In: IEEE CIT
4. Choi BG, Lee JY, Chung MY (2010) Adaptive binary negative-exponential backoff algorithm based on contention window optimization in ieee 802.11 wlan. KSII Trans Internet Inf Syst 4:896–909
5. Bharghavan V, Demers A, Shenker S, Zhang L (1994) Macaw: a media access protocol for wireless LAN's. In: SIGCOMM '94 Proceedings of the conference on communications architecture, protocols and applications. ACM, New York, pp 212–225
6. Haas ZJ, Deng J (2003) On optimizing the backoff interval for random access schemes. IEEE Trans Commun 51:2081–2090
7. Hastad J, Leighton T, Rogoff B (2001) Does the IEEE 802.11 MAC protocol work well in multihop wireless ad hoc networks. IEEE Commun Mag 39:130–137
8. IEEE: 802.11 (1997) IEEE standard for wireless LAN medium access control (MAC) and physical layer (phy) specifications
9. Deng J, Varshney PK, Haas ZJ (2004) A new backoff algorithm for the IEEE 802.11 distributed coordination function. Electrical Engineering and Computer Science, Paper 85. http://surface.syr.edu/eecs/85
10. Ke CH, Wei CC, Lin KW, Ding JW (2011) A smart exponential-threshold-linear backoff mechanism for IEEE 802.11 wlans. Int J Commun Syst 24:1033–1048
11. Kim DM, Kim SL (2013) An iterative algorithm for optimal carrier sensing threshold in random CSMA/CA wireless networks. IEEE Commun Lett 11:2076–2079
12. Kleinrock L, Tobagi FA (1975) Packet switching in radio channels: part i—carrier sensing multiple-access modes and their throughput-delay characteristics. IEEE Trans Commun 21:1400–1416
13. Kolar V, Razak S, Abu-ghazaleh NB, Mhnen P, Harras KA (2009) Analysis and impact of interactions in chains under CSMA protocol. Technical Reports, TR-NA-09-02
14. Luo T, Motani M, Srinivasan V (2012) Energy-efficient strategies for cooperative multichannel MAC protocols. IEEE Trans Mob Comput 11:553–566

15. Mehta S, Kwak KS (2010) Performance analysis of binary exponential backoff and improved backoff for WPAN. EURASIP J Wirel Commun Netw 2010. Art No 1. doi:10.1155/2010/6975735
16. Metcalfe RM, Boggs DR (1976) Ethernet: distributed packet switching for local computer networks. Commun ACM 19:395–404
17. Salameh HB, Shu T (2007) MarwanKrunz: adaptive crosslayer MAC design for improved energy-efficiency in multi-channel wireless sensor networks. Ad Hoc Netw 5:844–854
18. Thompson R, Zhou G, Lu L, Krishnamurthy S, Dong H, Qi X, Li Y, Keally M, Ren Z (2013) A self-adaptive spectrum management middleware for wireless sensor networks. Wireless Personal Commun 68:131–151
19. Wu Y, Stankovic JA, He T, Lu J, Lin S (2008) Realistic and efficient multi-channel communications in wireless sensor networks. In: IEEE INFOCOM, p 9
20. Ye SR, Tseng YC (2006) A multichain backoff mechanism for IEEE 802.11 wlans. Electr Eng Comput Sci 55:1613–1620
21. Zhang J, Zhou G, Huang C, Son SH, Stankovic JA (2007) Tmmac: an energy efficient multi-channel MAC protocol for ad hoc networks. In: IEEE ICC
22. Zhou G, Huang C, Yan T, He T, Stankovic JA, Abdelzaher TF (2006) Mmsn: multi-frequency media access control for wireless sensor networks. In: IEEE INFOCOM, p 7
23. Zhou G, Lu L, Krishnamurthy S, Keally M, Ren Z (2009) SAS: self-adaptive spectrum management for wireless sensor networks. In: IEEE ICCCN