Feng *et al. EURASIP Journal on Wireless Communications and Networking* (2016) 2016:60
DOI 10.1186/s13638-016-0557-9

EURASIP Journal on
Wireless Communications and Networking
a SpringerOpen Journal

RESEARCH Open Access

Analysis and optimization of delayed channel access for wireless cyber-physical systems

Li Feng[1], Jiguo Yu[2*], Xiuzhen Cheng[3] and Shengling Wang[4]

Abstract

In wireless cyber-physical systems (CPS), with a growing number of sensors being deployed, more and more data will be transmitted over wireless links. This essentially requires that wireless network protocols should be further studied and amended to improve their transmission efficiency. The delayed channel access (DCA) protocol is of great practical importance in improving channel utilization. DCA is an improved version of IEEE 802.11 distributed coordination function (DCF). In DCA, a node first waits for an extra delay before it enters the normal DCF procedure, so that more packets can be aggregated and transmitted upon each transmission opportunity. However, how the extra delay affects the performance of DCA has never been theoretically investigated. In this paper, we first propose a theoretical model to characterize the impact of a deterministic extra delay on collision probability, throughput, and medium access control (MAC) delay. With this model, we perform asymptotic analysis to calculate the optimal deterministic delay that can maximize the system throughput. We find that the system performance is significantly affected by the relationship between the deterministic delay and the number of nodes; and therefore, carefully choosing a suitable deterministic delay is crucial in improving channel utilization. The extensive ns2 simulations verify that our model is very accurate, and the theoretical optimal deterministic delay can make the system achieve the maximum system throughput. This study is very useful for better designing and implementing the packet aggregation technology that has been adopted by IEEE wireless networking standards including 802.11n and the latest 802.11ac.

Keywords: Cyber-physical systems, Wireless LAN, Delayed channel access, DCA, Performance

1 Introduction

Today, wireless networks such as IEEE 802.11 (WiFi), IEEE 802.15.1 (Bluetooth), and IEEE 802.15.4 (ZigBee) have been widely deployed. Wireless cyber-physical systems (CPS) [1–3] can fully utilize the already deployed wireless network to collect data and send control signals. Among a lot of scenarios, CPS requires an infrastructure-based wireless network for sensing, communication, and actuation, where wireless nodes monitor some aspects of the studied object (such as the environment), and relay the processed information to a central node [4]. For example, precision agriculture [5], patient-centric health-care applications and services [6], and smart home [7] provide centralized control of lighting, heating, ventilation, and air conditioning (HVAC), appliances, security locks of gates and doors, and other systems.

In the infrastructure-based wireless network, with more and more sensors being deployed, more and more data are required to be transmitted to the central node. In this situation, the conventional 802.11 distributed coordination function (DCF) protocol [8] (where each node is limited to send at most one data packet upon each transmission opportunity) is hard to meet the growing traffic transmission requirements. To improve the transmission efficiency, the frame aggregation technology (i.e., multiple packets are aggregated into a super-frame for one transmission) has been proposed and adopted by 802.11 EDCA [8], 802.11n [9–11], and the latest 802.11ac [12, 13]. 802.11 EDCA is an amendment of DCF. It defines a time duration

*Correspondence: jiguoyu@sina.com
[2]School of Information Science and Engineering, Qufu Normal University, 276826 Rizhao, Shandong, China
Full list of author information is available at the end of the article

called transmission opportunity limit (TXOP limit), in which a node can transmit multiple already arrived packets for each transmission. However, the advantage of the TXOP mechanism is not fully exploited in EDCA, because when a node wins the channel, maybe only a few packets arrive. To overcome the drawback of the TXOP mechanism, the delayed channel access (DCA) protocol [14] has been proposed, and it has attracted a great deal of attention [15–21]. The basic idea of DCA is that a node first waits for an extra random delay before it enters the conventional contention procedure in 802.11 DCF and EDCA, so that more packets can be aggregated for one transmission.

DCA defines three parameter thresholds to dynamically adjust the extra delay. Depending on these threshold settings, the extra delay varies significantly. So far, how to optimize these threshold settings has never been discussed. To solve this problem, we need to answer the fundamental question of DCA: what is the optimal extra delay? Obviously, if the extra delay is too short, only a few packets will be aggregated for one transmission and therefore, the advantage of the frame aggregation technology is still not fully exploited. On the other hand, if the extra delay is too long, it will cause a negative impact on quality of service for multimedia applications. This paper is devoted to studying the fundamental question of the optimal extra delay.

We note that no matter how we configure the parameter thresholds in DCA, it desires to make the extra random delay converge to a deterministic optimal delay. In this paper, we assume that the random delay in DCA is a deterministic constant and call such a DCA protocol the delayed-DCF protocol. This assumption not only simplifies the analysis but also enables us to focus on the fundamental question of DCA. Our contributions are threefold as follows:

- First, we develop a theoretical model to evaluate the performance of delayed DCF. This model can characterize the impact of the non-zero deterministic delay d on the collision probability, throughput, medium access control (MAC) delay mean, and MAC delay variance. We find that the system performance is significantly affected by the relationship between the deterministic delay d in millisecond (ms) and the number of nodes n. For example, the mean MAC access delay of each packet is almost equal to d ms when $d > n$, and is $O(n)$ ms otherwise.
- Second, with this model, we perform an asymptotic analysis to calculate the optimal deterministic delay that maximizes the system throughput.
- Third, we run extensive simulations to verify that the proposed model is very accurate and the optimal extra delay can maximize the system throughput.

The rest of this paper is organized as follows. Section 2 presents the related works. Section 3 outlines the DCF and delayed-DCF protocols. Section 4 proposes a theoretical model to evaluate the performance of the delayed-DCF protocol. Section 5 analyzes the optimal deterministic delay that maximizes the system throughput. Section 6 verifies the proposed theoretical model via ns2 simulations. Finally, Section 7 concludes this paper.

2 Related works

The aggregation technology is extremely useful in minimizing the number of transmissions and thereby improving the channel utilization and saving energy. For wireless routing in sensor networks, this technology enables each node to combine the data coming from different sources into a single packet of fixed size according to some aggregation operations such as logical and/or maximum or minimum, therefore reducing the traffic load greatly. The authors in [22–27] have made extensive studies on the technology for wireless sensor networks. For an infrastructure-based wireless network, this technology enables a node to aggregate its multiple own packets into a super-frame for one transmission, thereby improving the channel utilization.

In this section, focusing on the aggregation technology in the infrastructure-based wireless network, we first overview the packet aggregation mechanism in 802.11 EDCA [8] and 802.11n [9], and then point out its drawback; after that, we specify the DCA protocol [14], its enhanced versions [15, 16], and the latest works.

802.11 EDCA enhances 802.11 DCF by introducing service differentiation and packet aggregation. The packet aggregation mechanism can improve channel utilization, because it can reduce the header overhead of PHY- and MAC layers. In 802.11 EDCA, the traffic is divided into four access categories (ACs): voice traffic (AC_VO), video traffic (AC_VI), best-effort traffic (AC_BE), and background traffic (AC_BK). Each AC is configured with four contention parameters: arbitrary interframe spacing (AIFS), minimum and maximum contention windows ($\mathrm{CW}_{\mathrm{min}}$ and $\mathrm{CW}_{\mathrm{max}}$), and the TXOP limit. The TXOP limit provides a packet aggregation mechanism: the TXOP limit is a time duration during which a node can transmit multiple already arrived packets for one channel access. 802.11n extends the aggregation mechanism. In 802.11n, there are two types of aggregation: aggregated MAC service data unit (A-MSDU) and aggregated MAC protocol data unit (A-MPDU). The A-MSDU operation puts multiple MSDUs into a single MPDU, while A-MPDU operation further aggregates multiple MPDUs into one PLCP protocol data unit (PPDU).

In EDCA, differences in contention parameter values assigned to ACs create different channel access priorities. ACs (such as AC_VO and AC_VI) with high priority are assigned to smaller values of CWmin and CWmax and a larger value of TXOP limit, compared with ACs with low priority. However, high-priority flows just produce small aggregated sizes, because the small CW enables a flow to quickly transmit the arrived packets after a short waiting time. As a result, high-priority flows have high overhead and low channel utilization. Skordoulis et al. [16] gave such an example. Consider an HDTV application that has a mean rate of 19.2 Mbps and a constant MSDU size of 1500 bytes. Then, the packet interarrival time at the MAC layer is 625 microsecond (μs). Now, as the first packet arrives at the buffer, the node initiates a channel access using the AC_VI parameters. From [28], the mean MAC access delay (which is the interval between when the packet arrives at the buffer and when the packet arrives at the receiver) for the packet should be equal to about 241 μs. Then, by the time the second packet arrives, the first one has already been transmitted and there is no chance for the packets to be concatenated.

To overcome the drawback in EDCA that high-priority flows have low channel utilization, the DCA scheme, [14] proposed deferring the time of entering the conventional contention process, in order to wait for more packet arrivals for one transmission. However, the delay must be designed carefully. For example, a longer waiting time is likely to result in a larger aggregation size, but it could also leave the channel unnecessarily idle even when the packet queue is non-empty, as well as violate the quality of service requirement of a flow. To this end, DCA defines three thresholds: σ (which is a threshold of the aggregation size), τ (which is a threshold of the waiting delay), and α (which is a threshold of the idling time). The first parameter aims at maximizing the channel utilization, the second parameter considers the delay requirement of a flow, while the third parameter delimits the end time of a traffic burst. In DCA, a node will immediately enter the conventional contention process when the number of packets in the aggregation buffer is larger than σ, when the waiting delay of the first packet in the aggregation buffer is larger than τ, or when the idling time since the last packet reached the aggregation buffer is larger than α.

DCA can improve channel utilization. However, DCA might cause an adverse effect when it is applied to transmission control protocol (TCP) flows. TCP is a reliable, robust, and connection-oriented protocol for data delivery that is commonly used over the Internet because of its flexibility and adaptation to network status change. TCP adopts a congestion control algorithm for each flow. The algorithm defines a congestion window (CWND) which is used to indicate how many packets should be sent. By dynamically adjusting the CWND, TCP can well adapt to the network change. In the algorithm, the sender must wait for a number of segments that already had been sent, before it sets the CWND and carries on with the next set of segments. According to the receiving acknowledgement info, the sender can increase or decrease CWND. When DCA is applied, its first parameter σ hampers the adjustment of CWND. For example, if the network status is good, the CWND is limited by σ so that more packets cannot be sent; on the other hand, if the network status is bad, the parameter σ prevents the sender from transmitting fewer packets immediately.

To solve the aforementioned problem, the adaptive DCA (ADCA) [15] and the selective DCA (SDCA) [16] were proposed, where ADCA dynamically adjusts the DCA parameters so as to adapt to traffic characteristics, while SDCA selectively and prudently applies DCA to TCP traffic, in order to avoid adverse effect on the TCP performance. In addition, the authors in [20] pointed out that the extra delay in DCA will affect the quality of service for real-time applications. Our paper [17] investigated the relationship between the extra delay in DCA and the node number. All these related works studied the performance of DCA only via simulation. In contrast, this paper proposes a theoretical model to study how to model the impact of the extra delay and how to optimize the extra delay.

3 DCF and delayed DCF

3.1 IEEE 802.11 DCF protocol

The IEEE 802.11 DCF [8] is based on carrier sense multiple access with collision avoidance (CSMA/CA). DCF has two channel access mechanisms: the mandatory basic access mechanism and the optional request to send/clear to send (RTS/CTS) access mechanism.

In this paper, we consider the basic access mode only. With the help of Fig. 1a, we now describe the main procedure used in the basic access mode.

Before transmitting a packet, a node must sense the channel for at least a DCF interframe space (DIFS). During the DIFS time, if the channel is sensed idle, the node may begin the transmission process; if the channel is sensed busy, the node will defer access and enter a contention period.

During the contention period, the node employs the binary exponential backoff (BEB) algorithm to resolve collisions. In the BEB algorithm, a node initially generates a random backoff time uniformly distributed in [0, $CW_{min} - 1$], where CW_{min} is a given minimum CW size. Thereafter, the backoff counter decreases by one for each idle time slot and is suspended for each busy slot. The suspended backoff counter resumes after the channel is sensed idle for a DIFS period. When the backoff counter reaches zero, the node starts transmitting the head-of-line (HOL) packet at the beginning of the next time slot. For example, in Fig. 1a, node i first chooses a backoff time equal to 9 and starts counting down. When the backoff

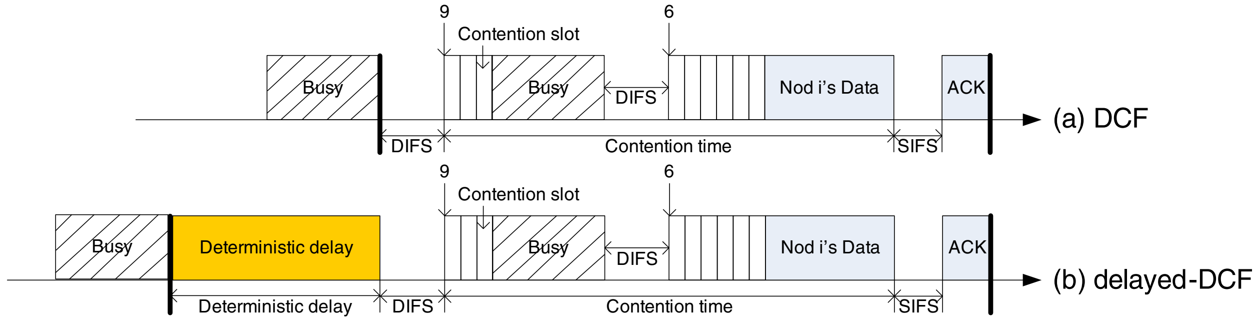

Fig. 1 Overview of **a** DCF and **b** delayed DCF

counter reduces to 6, node *i* suspends the counter because the channel is sensed busy and resumes the counter later after the channel is sensed idle for a DIFS period.

For each successful transmission, the sender will receive an acknowledgement (ACK) frame after a short interframe space (SIFS). If the node does not receive the ACK within a certain time (i.e., ACK timeout), it assumes that the data packet was not successfully received at the destination node and doubles the CW and repeats the above procedure. Doubling of the CW stops after the maximum window size $\mathrm{CW}_{\max}$ is reached. When a retransmission limit is reached, the sender discards the data packet. Note that according to the 802.11 DCF protocol, two consecutive data packet transmissions of a node are separated by at least a random time uniformly distributed in $[0, \mathrm{CW}_{\min} - 1]$.

3.2 Delayed DCF

Figure 1b illustrates the delayed DCF. Like DCF, a node in delayed DCF transmits at most one packet upon each transmission opportunity. However, unlike DCF, a node in delayed DCF always waits for a deterministic delay (denoted by *d* in this paper) before entering the subsequent DCF procedure. A special case is that when the deterministic delay, *d*, is equal to 0, the delayed-DCF protocol becomes the legacy DCF protocol.

The delayed DCF protocol actually adopts a mixed-type contention resolution method. One is the deterministic delay, which is independent of the channel status, and its counter are never suspended and will keep counting down once the counter is installed. The deterministic delay postpones the time that nodes contend for channel. Another is the backoff time, which is greatly affected by the channel status and therefore its counter will be suspended for a busy slot and resumed for subsequent idle slots. The backoff time increases as the contention becomes more intensive. The two types of delays objectively alleviate contention intensity.

This paper concerns the impact of the deterministic delay *d* on the performance of the subsequent DCF procedure, which is never investigated before.

4 Performance analysis

In this section, we propose a theoretical model to evaluate the performance of the delayed-DCF protocol. We first analyze the collision probability that governs all other performance metrics and then compute the throughput and the mean and variance of the MAC access delay.

We now introduce the terminologies and assumptions that we will use in our analysis: a packet transmission is said to be *finished* when the packet is either successfully received at the destination node or dropped due to reaching a retransmission limit; time is measured in MAC time slots unless explicitly indicated. Similar to [29], we assume that (1) all nodes are in saturation operation (i.e., each station always has a packet to transmit) and reside in a single-cell network (i.e., all stations are in the sensing range of each other); (2) the collision processes of the nodes can be decoupled; and (3) channel conditions are ideal so that transmission errors are a result of packet collision only.

4.1 Analysis of the collision probability

This section characterizes the collision probability and the attempt rate, which are governed by a fixed-point system described below.

Let γ denote the collision probability experienced by a tagged node on the condition that the buffer is not empty. Let β denote the attempt rate of each node (i.e., the ratio of the number of attempts in a generic slot) on the condition that the buffer is not empty, where the generic slot represents the time elapsed for one decrement of the backoff counter. According to the decoupling assumption (2), the tagged node will experience a collision if at least one of the remaining $n - 1$ nodes transmits, where n ($n \geq 2$) is the number of contending nodes. Therefore, we can express γ in terms of β as follows:

$$\gamma = 1 - (1-\beta)^{n-1}. \tag{1}$$

We now express β in terms of γ, following the approach in [29]. According to the BEB algorithm, a packet can undergo a maximum of M attempts, where each attempt is preceded by a backoff stage with a randomly selected backoff count η_k at stage k. The probability of $j+1$ attempts, where $j = 0, \ldots, M-1$, is given by

$$\delta(\gamma, j) = \begin{cases} (1-\gamma)\gamma^j, & j = 0, \cdots, M-2 \\ \gamma^{M-1}, & j = M-1, \end{cases} \tag{2}$$

where the upper term on the right is the probability that the packet suffers j collisions before success, and the lower term is the probability of either $M-1$ collisions before success or M collisions.

Let R and X be the number of attempts and the time (in generic slots) excluding DIFS, respectively, incurred by a packet transmission of the tagged node between when the node starts decreasing its deterministic delay and when its targeted packet transmission is finished. Then, X includes two components. The first one is the number of the generic slots (denoted by ξ) elapsed during the deterministic delay d. The second one is the number of the generic slots elapsed after the deterministic delay and before the targeted packet transmission of the tagged node is finished. From Eq. (2), we have

$$\begin{aligned} R &= j+1, \text{w.p. } \delta(\gamma, j), 0 \le j \le M-1, \\ X - \xi &= \sum\nolimits_{k=0}^{j} \eta_k, \text{w.p. } \delta(\gamma, j), 0 \le j \le M-1, \end{aligned} \tag{3}$$

where "w.p." means "with probability". In Eq. (3), η_k is uniformly distributed in $[0, \mathrm{CW}_k - 1]$ with mean $\overline{\eta}_k \triangleq b_k = (\mathrm{CW}_k - 1)/2$, where $\mathrm{CW}_k = 2^k \mathrm{CW}_0$ for $0 \le k \le m-1$ and $\mathrm{CW}_k = 2^m \mathrm{CW}_0$ for $m \le k \le M-1$; m determines the maximum backoff window size $\mathrm{CW}_{\max}$ (i.e., $\mathrm{CW}_{\max} = 2^m \mathrm{CW}_0$); and CW_0 is the minimum window size. Let $\overline{R}$, $\overline{X}$, and $\overline{\xi}$ denote the mean of R, X, and ξ, respectively. We have

$$\begin{aligned} \overline{R} &= \sum\nolimits_{j=0}^{M-1} (j+1)\delta(\gamma, j) \\ &= (1-r) + 2(1-r)r + \cdots \\ &= 1 - r + 2r - 2r^2 + \cdots \\ &= 1 + \gamma + \cdots + \gamma^{M-1}, \\ \overline{X} &= \overline{\xi} + \sum_{j=0}^{M-1} \sum_{k=0}^{j} b_k \delta(\gamma, j) \\ &= \overline{\xi} + b_0 \delta(\gamma, 0) + \sum_{k=0}^{1} b_k \delta(\gamma, 1) + \cdots \\ &= \overline{\xi} + b_0 + \gamma b_1 + \gamma^2 b_2 + \cdots + \gamma^{M-1} b_{M-1}. \end{aligned} \tag{4}$$

We now calculate $\overline{\xi}$. Let Ω be the length of a generic slot and $\overline{\Omega}$ be the mean of Ω. According to the definition of ξ, we have

$$\overline{\xi} = \frac{d}{\overline{\Omega}} \tag{5}$$

and hence, we only need to calculate $\overline{\Omega}$. The generic slot duration Ω depends on whether a slot is idle or interrupted by a successful transmission or a collision. We then define Ω as

$$\Omega = \begin{cases} \sigma & \text{w.p. } 1 - P_b, \\ T_s & \text{w.p. } P_s, \\ T_{\overline{s}} & \text{w.p. } P_{\overline{s}}, \end{cases} \tag{6}$$

where

$$\begin{aligned} P_b &= 1 - (1-\beta)^n = 1 - (1-\gamma)^{\frac{n}{n-1}}, \\ P_s &= n\beta(1-\beta)^{n-1} = n\left(1 - (1-\gamma)^{\frac{1}{n-1}}\right)(1-\gamma), \\ P_{\overline{s}} &= P_b - P_s. \end{aligned} \tag{7}$$

In Eq. (6), the MAC slot length $\sigma = 1$ slot $= 20$ µs. T_s and $T_{\overline{s}}$ are the mean time for a successful transmission and an unsuccessful transmission, respectively. The parameters T_s and $T_{\overline{s}}$ depend on packet payload length, SIFS, DIFS, and other protocol parameters. In Eq. (7), P_b denotes the probability of a busy slot, where $(1-\beta)^n$ is the probability that none of n nodes transmits packets in a slot; $1-(1-\beta)^n$ is the probability that at least one node transmits a packet and hence it is also the probability of a busy slot. P_s denotes the probability of a successful transmission from any of the n contending nodes. $P_{\overline{s}}$ denotes the probability of an unsuccessful transmission from any of the n contending nodes.

From Eq. (6), we can calculate $\overline{\Omega}$ by

$$\overline{\Omega} = (1-P_b)\sigma + P_s T_s + P_{\overline{s}} T_{\overline{s}}. \tag{8}$$

Now, applying the renewal reward theory, we have $\beta = \frac{\overline{R}}{\overline{X}}$. From Eqs. (4) and (5), β is given by

$$\beta = \frac{1 + \gamma + \cdots + \gamma^{M-1}}{\frac{d}{\overline{\Omega}} + b_0 + \gamma b_1 + \gamma^2 b_2 + \cdots + \gamma^{M-1} b_{M-1}}. \tag{9}$$

Note that when $d = 0$ in Eq. (9), β reduces to (1) in [29].

So far, we have expressed β in terms of γ in Eq. (9). Substituting β in Eq. (9) into Eq. (1), and solving the fixed-point equation with respect to γ, we can calculate the collision probability γ and then the attempt rate β.

4.2 Computation of throughput and delay

This section presents the formulae for the throughput and the mean and variance of the MAC access delay.

Throughput: For the per-node throughput, Γ, we adopt the expression derived in [29, 30], namely

$$\Gamma = \frac{P_s}{n}\frac{L}{\overline{\overline{\Omega}}}, \tag{10}$$

where L is the packet size in bits and $\frac{P_s}{n}$ is the per-node probability of a successful packet transmission.

Mean and variance of MAC access delay: We define the MAC access delay as the interval between when a packet enters the head-of-the-line of its queue and when the packet is successfully received at the destination node. Let D denote the MAC access delay and it consists of (1) the deterministic delay d and (2) the random time interval D_1 between when the deterministic delay ends and when the packet is successfully received at the destination node.

Let $\overline{D}$ and $\overline{D}_1$ denote the mean of D and D_1 , respectively. We have

$$\overline{D} = d + \overline{D}_1. \tag{11}$$

$\overline{D}_1$ can be calculated by Eq. (18) in [31] and is given as follows:

$$\begin{aligned}
\overline{D}_1 &= A_1 + B_1,\\
\text{where } A_1 &= \frac{1-\gamma}{(1-\gamma^M)}\sum\nolimits_{i=0}^{M-1}\gamma^i\left\{\theta_1\sum\nolimits_{k=0}^{i}\overline{\eta}_k + iT_{\bar{s}}\right\},\\
B_1 &= T_s - T_{\text{ACK}},\\
T_{\text{ACK}} &= \text{the transmission time of an ACK packet,}
\end{aligned}$$

where θ_1 is defined in Eq. (13).

Let $\text{Var}(D)$ denote the variance of D. We have

$$\text{Var}(D) = \text{Var}(D_1). \tag{12}$$

$\text{Var}(D_1)$ can be calculated by Eq. (19) in [31] and is given as follows:

$$\begin{aligned}
\text{Var}(D_1) &= \frac{1-\gamma}{(1-\gamma^M)}\sum\nolimits_{i=0}^{M-1}\gamma^i\left\{A_2^i + B_2^i\right\},\\
\text{where } A_2^i &= \sum\nolimits_{k=0}^{i}\left(\overline{\eta}_k\theta_3 + (\theta_1)^2\text{Var}(\eta_k)\right),\\
B_2^i &= \left(\theta_1\sum\nolimits_{k=0}^{i}\overline{\eta}_k + iT_{\bar{s}} - A_1\right)^2,
\end{aligned}$$

where θ_1 and θ_3 are defined in Eq. (13).

$$\begin{aligned}
q &= (n-1)\beta(1-\beta)^{n-2}, \\
\theta_1 &= \sigma + \theta_2,\\
\theta_2 &= (qT_s + (\gamma - q)T_{\bar{s}})(1-\beta),\\
\theta_3 &= \left(q(T_s-\theta_2)^2 + (\gamma-q)(T_{\bar{s}}-\theta_2)^2\right)(1-\beta)\\
&\quad + (1-\gamma(1-\beta))(\theta_2)^2.
\end{aligned} \tag{13}$$

5 Optimal deterministic delay

In this section, we seek the optimal deterministic delay so as to maximize the system throughput. Below, we first find the optimal attempt rate and then calculate the optimal deterministic delay.

5.1 Optimal attempt rate

This section finds the optimal attempt rate, β_{opt}, that maximizes the system throughput.

From Eqs. (10) and (8), the system throughput $n\Gamma$ is given by

$$\begin{aligned}
n\Gamma &= \frac{P_sL}{(1-P_b)\sigma + P_sT_s + P_{\bar{s}}T_{\bar{s}}}\\
&= \frac{L}{T_s + \frac{(1-P_b)\sigma + P_{\bar{s}}T_{\bar{s}}}{P_s}}.
\end{aligned} \tag{14}$$

Then, from Eq. (14), to maximize $n\Gamma$, we just need to maximize $h(\beta)$:

$$h(\beta) \triangleq \frac{(1-P_b)\sigma + P_{\bar{s}}T_{\bar{s}}}{P_s}. \tag{15}$$

Setting the first-order derivative of Eq. (15) in terms of β to 0, we obtain the equation

$$\begin{aligned}
1 - n\beta &= \eta(1-\beta)^n, \\
\text{where } \eta &= 1 - \frac{\sigma}{T_{\bar{s}}}.
\end{aligned} \tag{16}$$

Let $\varphi \triangleq \lim_{n\to\infty} n\beta$ be the asymptotic aggregate attempt rate. Assume that φ exists and apply the Poisson approximation to Eq. (16). We have

$$\begin{aligned}
1-\varphi &= \eta e^{-\varphi}\\
1-\varphi &= \frac{\eta}{e}e^{-\varphi+1}\\
(\varphi-1)e^{\varphi-1} &= -\frac{\eta}{e}.
\end{aligned}$$

Let $W_0(\cdot)$ be one branch of the Lambert $W(z)$ function [32] and $W(z)e^{W(z)} = z$ for any complex number z. Then,

$$\varphi - 1 = W_0\left(-\frac{\eta}{e}\right) \text{ or } W_{-1}\left(-\frac{\eta}{e}\right).$$

Note that $W_0\left(-\frac{\eta}{e}\right) > -1$ and $W_{-1}\left(-\frac{\eta}{e}\right) < -1$ for $-\frac{\eta}{e} \in \left(\frac{-1}{e}, 0\right)$. Then, the optimal φ, φ_{opt}, is

$$\varphi_{\text{opt}} = W_0\left(-\frac{\eta}{e}\right) + 1.$$

Because $\varphi_{\text{opt}} \approx n\beta_{\text{opt}}$ for a given n, the optimal attempt rate, β_{opt}, is

$$\beta_{\text{opt}} \approx \frac{1}{n}\left[W_0\left(-\frac{\eta}{e}\right) + 1\right]. \tag{17}$$

Remark. Under the assumption that each node just transmits one packet upon one transmission opportunity, Eq. (17) is derived. When the packet aggregation technology is adopted, Eq. (17) can hold true as well and what we need is to recalculate η (or $T_{\bar{s}}$). For example, assume that each node can fill K packets with length L bytes into the payload filed of the 802.11 MAC frame and transmit the superframe upon one transmission opportunity. Then, we can keep all deductions from Eqs. (14) to (17) unchanged, except that in Eq. (14), we replace L by KL, calculate T_s by KL, and set $T_{\bar{s}} = T_s$ if the basic access mechanism is adopted and set $T_{\bar{s}} = T_{\text{rts}} + T_{\text{sifs}} + T_{\text{cts}}$ if the RTS/CTS access mechanism is adopted, where T_{rts}, T_{sifs}, and T_{cts}, respectively, are the RTS transmission time, the SIFS time, and the CTS transmission time.

5.2 Optimal deterministic delay

According to the delayed DCF, a node always first waits for a deterministic delay and then enters the subsequent DCF procedure. Therefore, the deterministic delay is closely connected to the attempt rate. In this section, we calculate the optimal deterministic delay, d_{opt}, that achieves optimal attempt rate β_{opt}.

From Eqs. (17) and (1), the optimal attempt rate β_{opt} and the optimal collision probability γ_{opt} are given by Eq. (18):

$$\begin{cases} \beta_{\text{opt}} = \frac{1}{n}\left[W_0\left(-\frac{n}{e}\right) + 1\right] \\ \gamma_{\text{opt}} = 1 - \left(1 - \beta_{\text{opt}}\right)^{n-1}. \end{cases} \tag{18}$$

Note that β_{opt}, γ_{opt}, and d_{opt} are constrained by Eq. (9). d_{opt} is expressed as follows:

$$d_{\text{opt}} = \overline{\Omega}(\beta_{\text{opt}})\left[\frac{1 + \gamma_{\text{opt}} + \cdots + \gamma_{\text{opt}}^{M-1}}{\beta_{\text{opt}}} - \left(b_0 + \gamma_{\text{opt}} b_1 + \gamma_{\text{opt}}^2 b_2 + \cdots + \gamma_{\text{opt}}^{M-1} b_{M-1}\right)\right], \tag{19}$$

where $\overline{\Omega}(\cdot)$ is given by Eq. (8).

6 Model verification

In the previous two sections, we theoretically analyze the performance of the delayed-DCF protocol and the optimal deterministic delay that maximizes the system throughput. In this section, we verify these theoretical results using the 802.11 simulator in ns2 version 2.28 [33] with some modifications. The purpose of these modifications is to add a deterministic delay d into the traditional DCF so as to be consistent with the delayed-DCF protocol.

We consider a one-hop star network with an access point (AP) and n saturated nodes, where the AP only acts as the receiver of data packets from all nodes. We present the theoretical results under the assumption of $T_s = T_{\bar{s}}$. For practical networks, this assumption does not necessarily hold. However, many analytical studies, including [30, 34] and [29], adopt this assumption, and it has also been adopted by the developers of the ns2 simulation tool [33]. The assumption is equivalent to assuming that ACK packets are transmitted at the basic rate, and the ACK timeout after a collision matches the guard time observed by non-colliding nodes.

In our simulations, we use the DumbAgent routing protocol and set the simulation time to 100 s. We assume that each node and the AP have an infinite buffer and hence, set the buffer size to 1000 packets. The default parameter values shown in Table 1 are set in accordance with 802.11b. In addition, in all figures, we use "sim" and "ana" to denote the simulation and analysis results, respectively.

Below, we first demonstrate the accuracy of our performance model in Section 6.1 and then verify our optimization result in Section 6.2.

6.1 Performance evaluation

In this subsection, we study the impact of d on the throughput, the collision probability, the mean and standard deviation of the MAC access delay when the number of nodes n varies from 4 to 30, where $d = 5$, 10 ms. It has been proven in [31] that for a one-hop WLAN with n contending nodes and $d = 0$ ms, the mean MAC delay of a packet is $O(n)$ ms. We will show that the system performance of the delayed-DCF protocol varies significantly, relying on the relationship between d and n. The main observation is that the dominant component of the MAC access delay is the deterministic delay of d ms when $d > n$ and is the random delay in DCF of $O(n)$ ms otherwise. In our simulations, we change n from 4 to 30 and set $d = 5$, 10 ms, in order to illustrate this observation.

Figure 2 plots the collision probability versus the number of nodes, where the theoretical results are calculated

Table 1 Default parameter settings used in this paper

CW_0	32	Header	241 µs	= Mheader + Pheader + RouteHeader
m/M	5/7	T_s	940 µs	= Header + L_{tm} + SIFS +**δ**+ ACK +**δ**+ DIFS
σ	20 µs	$T_{\bar{s}}$		= T_s
δ	0 µs	L_{tm}	335 µs	= 460 bytes @ R_{data}
SIFS	10 µs	ACK	304 µs	= 24 bytes @ R_{basic}+ 14 bytes @ R_{basic}
DIFS	50 µs	Mheader	20 µs	= 24 bytes @ R_{data}+ 4 bytes @ R_{data}
R_{data}	11 Mbps	Pheader	192 µs	= 24 bytes @ R_{basic}
R_{basic}	1 Mbps	RouteHeader	29 µs	= 40 bytes @ R_{data}

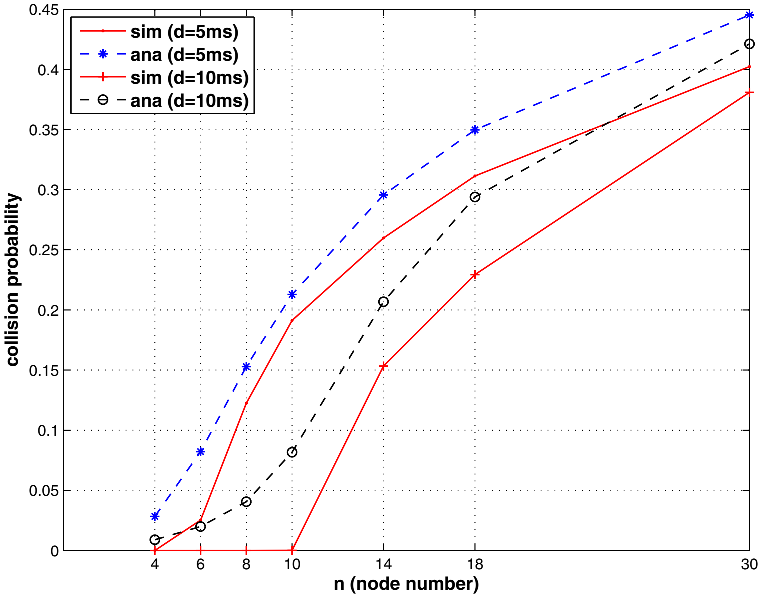

Fig. 2 The collision probability versus the number of nodes

by Eqs. (9) and (1). From this figure, we can see that the collision probability increases as n increases when $d = 5, 10$ ms. However, for each n, the collision probability when $d = 5$ ms is obviously larger than that when $d = 10$ ms. The reason is that the larger d greatly alleviates the contention intensity, leading to a lower collision probability. Particularly, when n increases from 4 to 10, the simulated collision probability when $d = 5$ ms increases from 0.02 to 0.22, whereas it is zero when $d = 10$ ms.

Figure 3 plots the per-node throughput versus the number of nodes when $d = 5, 10$ ms, where the theoretical results are calculated by Eq. (10). From this figure, we observe that (a) the per-node throughput when $d = 5$ ms, decreases as n increases from 4 to 30, while the per-node throughput when $d = 10$ ms, first remains unchanged for $n \leq 10$, and then decreases for $n > 10$; (b) the throughput curve when $d = 5$ ms is obviously above that when $d = 10$ ms for $n \leq 10$, but the two curves almost overlap for $n > 10$. The reasons are as follows: the delay and hence the per-node throughput is governed by d for $d > n$ and are governed by n otherwise. Further, for each d, the theoretical curve (labeled with "ana") closely matches the simulation curve (labeled with "sim"), manifesting that our theoretical model is very accurate.

Figure 4 plots the mean MAC access delay versus the number of nodes, where the theoretical results are calculated by Eq. (11). From this figure, the MAC access delay when $d = 10$ ms is almost equal to 10 ms and is obviously less than that when $d = 5$ ms for $n < 10$; however, it is almost equal and is $O(n)$ ms for each $n > 10$ when $d = 5, 10$ ms. This manifests that the dominant component of the MAC access delay is the deterministic delay d ms when $d > n$ and is the random delay in DCF of $O(n)$ ms otherwise.

Figure 5 plots the standard deviation of the MAC access delay versus the number of nodes, where the theoretical results are calculated by Eq. (12). From this figure, we can see that the standard deviation increases as n increases when $d = 5, 10$ ms. However, the standard deviation when $d = 5$ ms is obviously larger than when $d = 10$ ms for each n. Particularly, the standard deviation when $d = 10$ ms for $n < 10$ is almost zero since the MAC access delay is almost a constant and is equal to 10 ms under this case.

Finally, from all these figures, we can see that all theoretical results, except the theoretical collision probability which slightly deviates from the simulated value, well match the corresponding simulated results, indicating that our model is very accurate.

6.2 Optimal deterministic delay

In this subsection, we verify the accuracy of the theoretical optimal deterministic delay and illustrate that the optimal deterministic delay can significantly improve the system performance. To this end, we calculate the optimal deterministic delay and compare the throughput, the collision probability, the mean and standard deviation of the MAC access delay when the fixed deterministic delay $d = 5$ ms, and the optimal deterministic delay $d = d_{\text{opt}}$ are used, where d_{opt} is measured in microseconds and is calculated by Eq. (19).

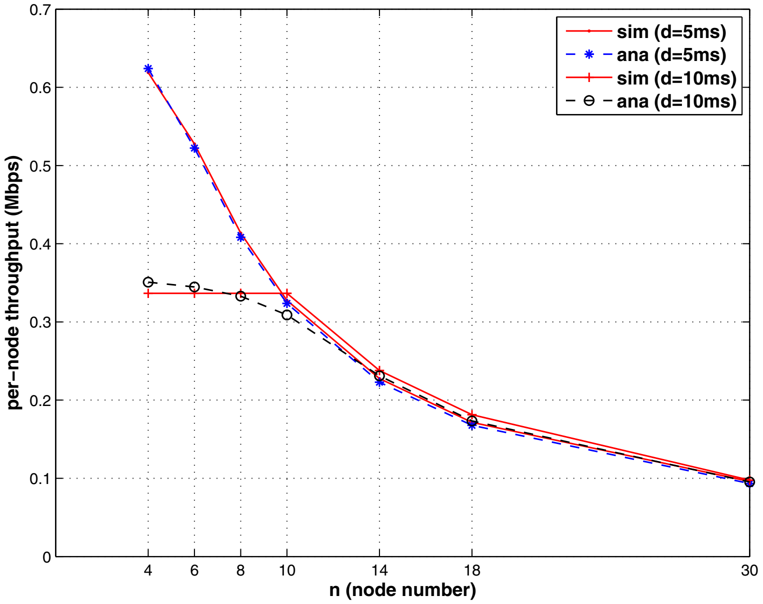

Fig. 3 The per-node throughput versus the number of nodes

Figure 6 plots the optimal deterministic delay versus the number of nodes when the packet size $L = 460$ and 1000 bytes, where the theoretical results are calculated by Eq. (19). From this figure, we can see that the optimal delay increases as n increases. This is because that a larger node number means higher contention intensity; a larger delay will reduce the number of the nodes that simultaneously attempt to contend for channel. On the other hand, for the same node number, the optimal delay when the packet size is large is higher than that when the packet size is small. The reason is that a larger packet size means a larger traffic load when the node number is the same; a larger delay can reduce the number that nodes attempt to access channel in a unit time, thereby reducing the contention intensity.

Figure 7 plots the collision probability versus the number of nodes, where the theoretical results are calculated by Eqs. (9) and (1). From this figure, when $d = 5$ ms,

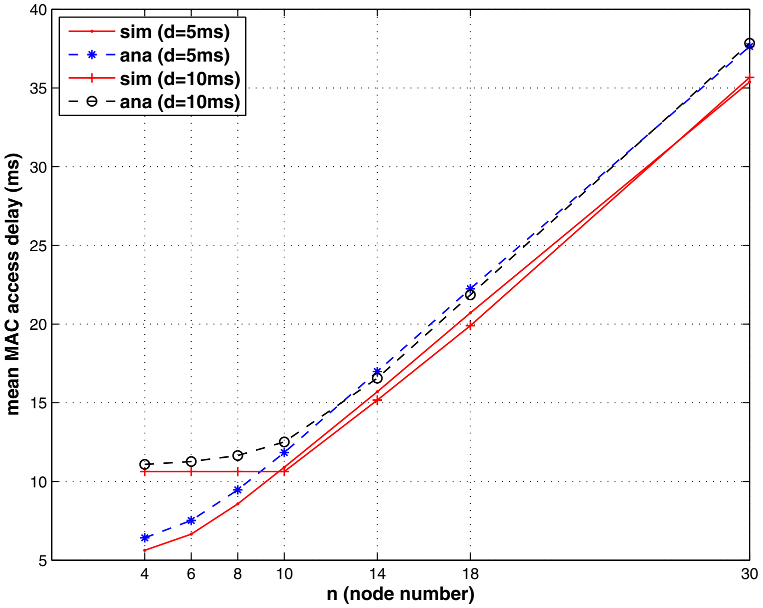

Fig. 4 The mean MAC access delay versus the number of nodes

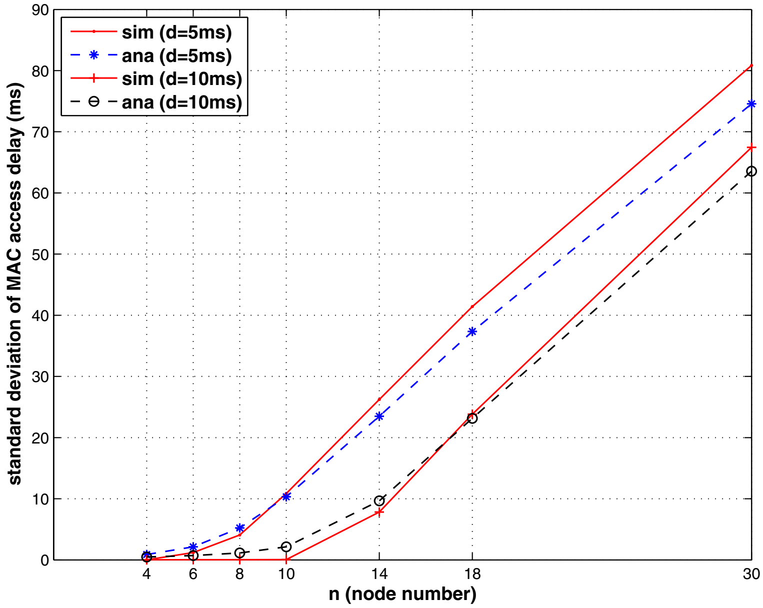

Fig. 5 The standard deviation of the MAC access delay versus the number of nodes

we can see that the collision probability significantly increases from 0 to 0.45 as n increases from 4 to 30. In contrast, when $d = d_{\text{opt}}$, we can see that the collision probability is always below 0.1, regardless of how n varies.

Figure 8 plots the system throughput $n\Gamma$ versus the number of nodes, where the theoretical results are calculated by Eq. (10). From this figure, when $d = 5$ ms, we can see that the system throughput first decreases from 4.8 to 5 Mbps as n increases from 4 to 6 and then decreases from 5 to 4.2 Mbps as n continues to increase from 6 to 30. The reason that the system throughput has a maximum value when $n = 6$ is as follows: the $d = 5$ ms approximates the optimal deterministic delay. In contrast, when $d = d_{\text{opt}}$, we can see that the system throughput slightly fluctuates around

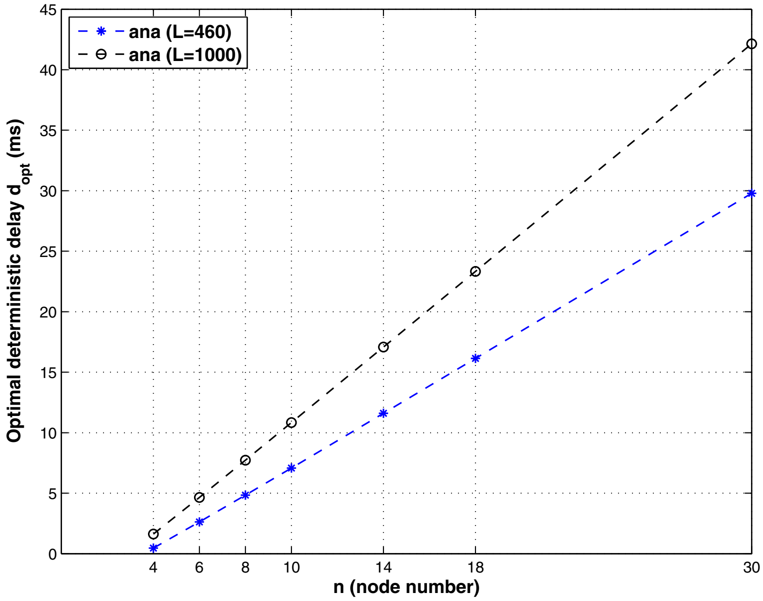

Fig. 6 Optimal deterministic delay versus the number of nodes

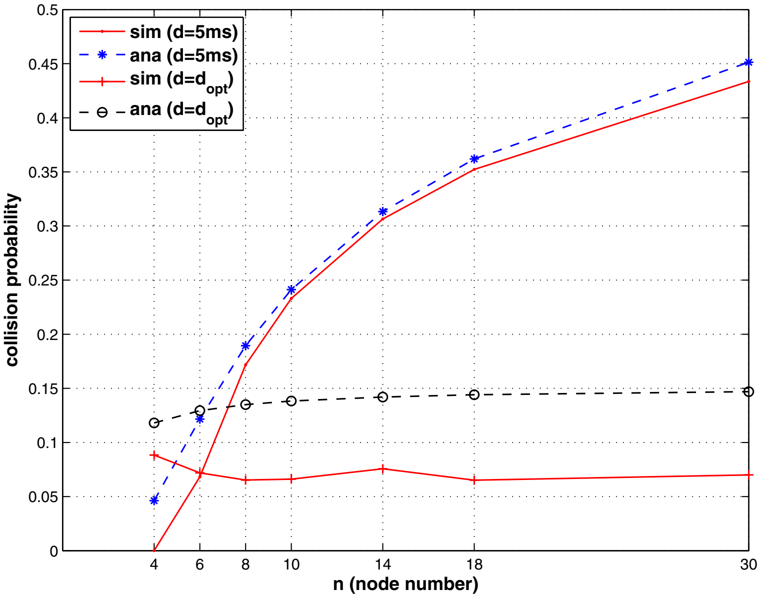

Fig. 7 The collision probability versus the number of nodes

5.1 Mbps, which is always larger than that when $d = 5$ ms.

Figure 9 plots the mean MAC access delay versus the number of nodes, where the theoretical results are calculated by Eq. (11). From this figure, the MAC access delay when $d = d_{\text{opt}}$ is always not larger than that when $d = 5$ ms.

Figure 10 plots the standard deviation of the MAC access delay versus the number of nodes, where the theoretical results are calculated by Eq. (12). From this figure, when $d = 5$ ms, we can see that the standard deviation significantly increases from 0 to 115 ms as n increases from 4 to 30. In contrast, when $d = d_{\text{opt}}$, we can see that the standard deviation is always below 5 ms, regardless of how n varies.

Finally, from all these figures, we can see that all theoretical results well match the corresponding simulated

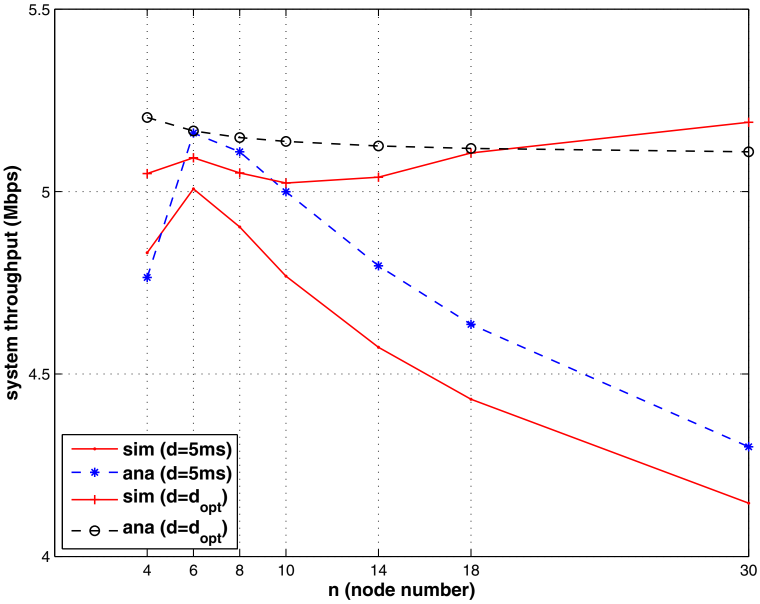

Fig. 8 The system throughput versus the number of nodes

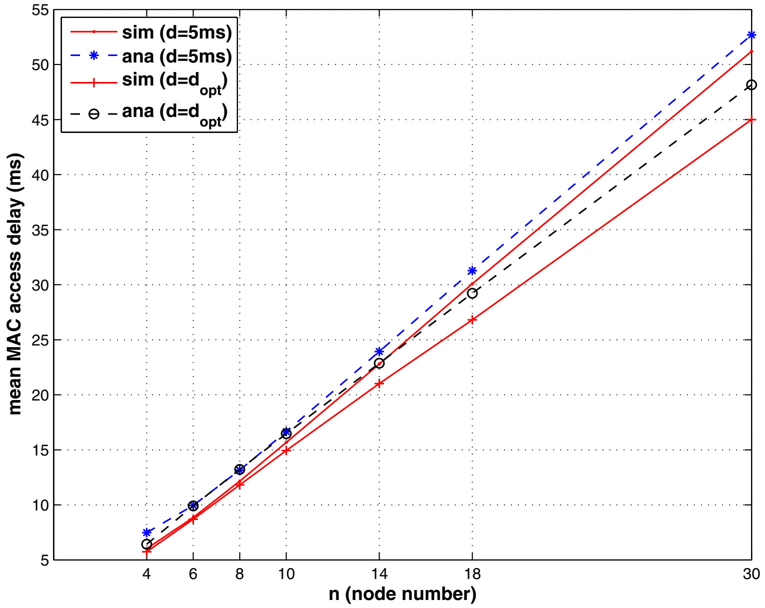

Fig. 9 The mean MAC access delay versus the number of nodes

results, indicating that the theoretical optimal deterministic delay is accurate.

7 Conclusions

With more and more data being transmitted in wireless cyber-physical systems, wireless network protocols should be further studied and amended to improve their transmission efficiency. This paper proposes a theoretical model to study the performance of the delayed-DCF protocol. With this model, we can accurately characterize the impact of the deterministic delay parameter in the protocol on the system performance and optimize the deterministic delay parameter setting. The delayed-DCF protocol is a special case of the DCA protocol, whose idea is to introduce an extra random delay so that more packets can be backlogged and transmitted upon each transmission opportunity. The study is very helpful for further modeling of the DCA protocol and for bet-

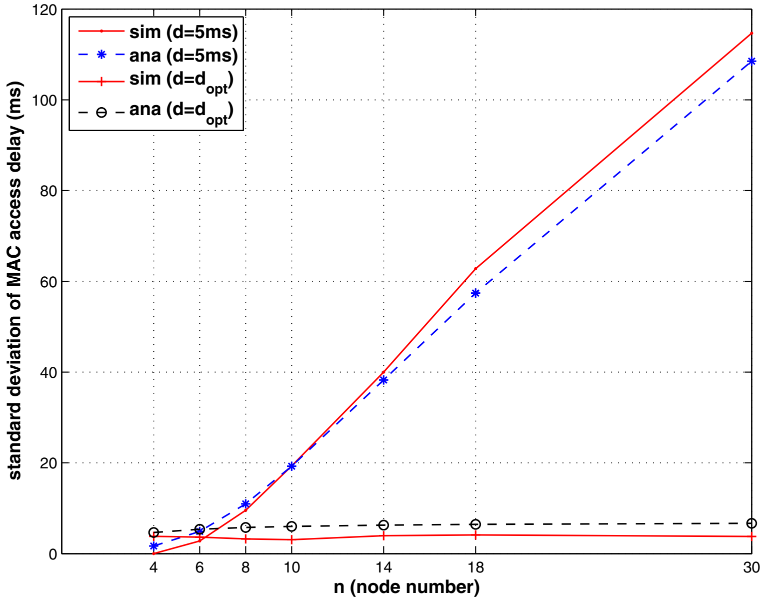

Fig. 10 The standard deviation of the MAC access delay versus the number of nodes

ter designing and implementing of the packet aggregation technology that has been widely adopted by IEEE wireless networking standards including 802.11 n and the latest 802.11 ac.

Competing interests

The authors declare that they have no competing interests.

Acknowledgements

This work is partially supported by the Macao Science and Technology Development Fund under Grants (No. 013/2014/A1 and No. 104/2014/A3), the NNSF of China under Grant 61373027, and NSF of the Shandong Province under Grant ZR2012FM023.

Author details

[1] Faculty of Information Technology, Macau University of Science and Technology, Macau, China. [2] School of Information Science and Engineering, Qufu Normal University, 276826 Rizhao, Shandong, China. [3] Department of Computer Science, the George Washington University, Washington, DC, USA. [4] College of Information Science and Technology, Beijing Normal University, 100875 Beijing, China.

Received: 4 November 2015 Accepted: 8 February 2016
Published online: 24 February 2016

References

1. EA Lee, Cyber physical systems: design challenges, Technical Report No. UCB/EECS-2008-8. Available at: http://www.eecs.berkeley.edu/Pubs/TechRpts/2008/EECS-2008-8.pdf. Accessed 2015
2. W Kang, K Kapitanova, SH Son, RDDS: A real-time data distribution service for cyber-physical systems. IEEE Trans. Industrial Inform. **8**(2), 393–405 (2012)
3. EA Lee, The past, present and future of cyber-physical systems: a focus on models. Sensors. **15**, 4837–4869 (2015)
4. S Cheng, Z Cai, J Li, X Fang, in *INFOCOM*. Drawing dominant dataset from big sensory data in wireless sensor networks, (2015), pp. 531–539
5. https://en.wikipedia.org/wiki/Precision_agriculture. Accessed 2015
6. Y Zhang, M Qiu, C Tsai, M Hassan, A Alamri, Health-CPS: healthcare cyber-physical system assisted by cloud and big data. IEEE Syst. J. Accepted for publication
7. https://en.wikipedia.org/wiki/Home_automation. Accessed 2015
8. IEEE Std. 802.11-2007, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, June 2007
9. IEEE 802.11 WG, IEEE P802.11n-2009: Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Amendment 5: Enhancements for Higher Throughput, IEEE Standards Association Std., Oct. 2009.
10. Y Xiao, IEEE 802.11n: enhancements for higher throughput in wireless LANs. Wirel. Commun. IEEE. **12**(6), 82–91 (2005)
11. E Perahia, IEEE 802.11n development: history, process, and technology. Commun. Mag. IEEE. **46**(7), 48–55 (2008)
12. M Gong, B Hart, S Mao, Advanced wireless LAN technologies: IEEE 802.11ac and beyond. ACM Mobile Comput. Commun. Rev. (MC2R). **18**(4), 48–52 (2014)
13. F Siddiqui, S Zeadally, K Salah, Gigabit wireless networking with IEEE 802.11ac: technical overview and challenges. J. Netw. **10**(3), 164–171 (2015)
14. L Changwen, AP Stephens, Delayed channel access for IEEE 802.11e based WLAN. IEEE Int. Conf. Commun., 4811–4817 (2006)
15. D Skordoulis, Q Ni, G Min, K Borg, Adaptive delayed channel access for IEEE 802.11n WLANs. 4th IEEE Int. Conf. Circ. Syst. Commun., 167–171 (2008)
16. D Skordoulis, Q Ni, C Zarakovitis, A selective delayed channel access (SDCA) for the high-throughput IEEE 802.11n. IEEE Wirel. Commun. Netw. Conf. (WCNC 2009), 1–6 (2009)
17. Q Zhao, Z Ma, H Dai, Performance evaluation of the delayed-DCF scheme in wireless LANs. Int. J. Future Comput. Commun. **2**(5), 391–394 (Oct. 2013)
18. Z Abichar, JM Chang, Group-based medium access control for IEEE 802.11n wireless LANs. IEEE Transac. Mobile Comput. **12**(2), 304–317 (2013)
19. G Marta, C Davide, CA Fabio, AJ Singh, MP Elgaard, A multi-QoS aggregation mechanism for improved fairness in WLAN. Veh. Technol. Conf. (VTC Fall), 2013 IEEE 78th, 1–5 (2013)
20. B Maqhat, M Baba, R Rahman, A Saif, Scheduler algorithm for IEEE802.11n wireless LANs. Int. J. Future Comput. Commun. **3**(4), 224–226 (2014)
21. O Sharon, Y Alpert, The combination of QoS, aggregation and RTS/CTS in very high throughput IEEE 802.11ac networks. Physical Commun. **15**, 25–45 (Jun. 2015)
22. J Li, S Cheng, H Gao, Z Cai, Approximate physical world reconstruction algorithms in sensor networks. IEEE Transac. Parallel and Distrib. Syst. **25**(12), 3099–3110 (2014)
23. Z Cai, R Goebel, G Lin, Size-constrained tree partitioning: approximating the multicast k-tree routing problem. Theor. Comput. Sci. **412**(3), 240–245 (2011)
24. S Cheng, Z Cai, J Li, Curve query processing in wireless sensor networks. IEEE Transac. Veh. Technol. **64**(11), 5198–5209 (2015)
25. X Zheng, J Li, H Gao, Z Cai, in *MOBIHOC*. Capacity of wireless networks with multiple types of multicast sessions, (2014), pp. 135–144
26. Z Cai, Z-Z Chen, G Lin, A 3.4713-approximation algorithm for the capacitated multicast tree routing problem. Theor. Comput. Sci. **410**(52), 5415–5424 (2008)
27. Z Cai, G Lin, G Xue, in *COCOON*. Improved approximation algorithms for the capacitated multicast routing problem, (2005), pp. 136–145
28. IEEE P802.11.2, Draft 1.0, Recommended practice for the evaluation of 802.11 wireless performance, IEEE-802.11 WG, April 2007
29. A Kumar, E Altman, D Miorandi, M Goyal, New insights from a fixed point analysis of single cell IEEE 802.11 WLANs. IEEE/ACM Transac. Netw. **15**(3), 588–601 (2007)
30. G Bianchi, Performance analysis of the IEEE 802.11 distributed coordination function. IEEE J. Sel. Areas. Commun. **18**(3), 535–547 (2000)
31. T Sakurai, HL Vu, Access delay of the IEEE 802.11 MAC protocol under saturation. IEEE Transac. Wirel. Commun. **6**(5), 1702–1710 (2007)
32. RM Corlessa, GH Gonnet, DEG Hare, DJ Jeffrey, DE Knuth, On the Lambert W function. Adv. Comput. Math. **5**, 329–359 (1996)
33. http://www.isi.edu/nsnam/ns/ns-build.html. Accessed 2015
34. D Malone, K Duffy, D Leith, Modeling the 802.11 distributed coordination function in non-saturated heterogeneous conditions. IEEE/ACM Transac. Netw. **15**(1), 159–172 (2007)