Incomplete cooperation-based service differentiation in WLANs

Liqiang Zhao¹*, Li Cong¹, Wei Ding², Xinheng Wang³ and Hailin Zhang¹

¹ State Key Laboratory of Integrated Services Networks, Xidian University, China
² Centre for Wireless Network Design, University of Bedfordshire, U.K.
³ Institute of Advanced Telecommunications, School of Engineering, Swansea University, U.K.

ABSTRACT

In the IEEE 802.11 wireless LAN (WLAN), the fundamental medium access control (MAC) mechanism—distributed coordination function (DCF), only supports best-effort service, and is unaware of the quality-of-service (QoS). IEEE 802.11e enhanced distributed channel access (EDCA) supports service differentiation by differentiating contention parameters. This may introduce the problem of non-cooperative service differentiation. Hence, an incompletely cooperative EDCA (IC-EDCA) is proposed in this paper to solve the problem. In IC-EDCA, each node that is cooperative a priori adjusts its contention parameters (e.g., the contention window (CW)) adaptively to the estimated system state (e.g., the number of competing nodes of each service priority). To implement IC-EDCA in current WLAN nodes, a frame-analytic estimation algorithm is presented. Moreover, an analytical model is proposed to analyze the performance of IC-EDCA under saturation cases. Extensive simulations are also carried out to compare the performances of DCF, EDCA, incompletely cooperative game, and IC-EDCA, and to evaluate the accuracy of the proposed performance model. The simulation results show that IC-EDCA performs better than DCF, EDCA, and incompletely cooperative game in terms of system throughput or QoS, and that the proposed analytical model is valid. Copyright © 2010 John Wiley & Sons, Ltd.

KEYWORDS

WLAN; MAC; QoS; service differentiation; game theory

*Correspondence

Liqiang Zhao, State Key Laboratory of Integrated Services Networks, Xidian University, China.
E-mail: lqzhao@mail.xidian.edu.cn

1. INTRODUCTION

In recent years, wireless LANs (WLANs) have been widely used as one of the essential technologies to provide broadband wireless access, and the performance analysis and improvement of WLANs have attracted much research interests [1]. IEEE 802.11 is a widely adopted WLAN standard, and its fundamental medium access control (MAC) mechanism—distributed coordination function (DCF)—is based on carrier sense multiple access with collision avoidance (CSMA/CA), a typical contention-based MAC protocol. DCF only supports best-effort service, and does not ensure any quality-of-service (QoS) requirement. However, there is an increasing demand that multimedia services with different QoS requirements be supported by WLANs. Different multimedia services have different QoS requirements with respect to bandwidth, delay, jitter, and packet-loss-rate. In order for WLANs to support multimedia applications, the MAC protocol must support a degree of service differentiation. However, in DCF all the traffic classes have the same contention parameters. Hence, all the services have the same priority when accessing the channel. One possible solution is to provide a good priority scheme by differentiating the contention parameters, thereby achieving service differentiation [2–7]. The IEEE 802.11e standard introduces enhanced distributed channel access (EDCA) to support QoS applications. In EDCA, the traffic which is assigned a smaller CW has a higher priority to contend for the channel.

One of the limitations of the current 802.11e is that it does not consider the impact of varying network conditions, e.g., channel conditions and network load. Some pieces of work are attempting to address these challenges, e.g., channel-dependent packet-level tuning (ChaPLeT), adaptive EDCF (AEDCF), and adaptive fair EDCA, three representative enhanced methods [5]. ChaPLeT focuses on improving service differentiation by configuring three selected parameters, i.e., fragmentation threshold, persistence factor, and defer countdown, based on monitored channel conditions. However, ChaPLeT does not consider the impact of network load. AEDCF focuses on improving throughput
under high network load by adapting the CW appropriately to reduce the number of collisions. However, AEDCF does not consider the impact of traffic loads with different priorities. Adaptive fair EDCAF focuses on improving node performance, e.g., fairness, by adapting two parameters, the magnification factor and the backoff mechanism. In a word, all the above protocols have not considered the problem of non-cooperative service differentiation.

Different traffics have different requirements on QoS. Generally, important QoS metrics includes bounds on bandwidth, delay, jitter, and packet-loss-rate. Real-time applications (e.g., voice and video) have strict delay requirements, but are not very sensitive to packet loss. Best-effort applications (e.g., file transfer) are sensitive to packet loss, but can tolerate large delays. Hence, to ensure its QoS, the real-time traffic wants to get a higher priority to access the channel than the best-effort traffic. Based on the service differentiation mechanism, EDCA can support multimedia services, i.e., the high priority traffic (e.g., voice and video applications) with a lower CW will enjoy greater bandwidth and less delay, compared to the low priority traffic (e.g., file transfer). That is to say, the service differentiation-based EDCA is unfair when supporting multimedia services.

In EDCA, due to the non-cooperative behaviors of nodes, the nodes cannot adjust their contention strategies to the various types of services and network loads, which may cause several problems in the above unfair manner. Firstly, a smaller CW may cause more collisions, and it will significantly affect the system performance. Secondly, the low priority traffic with a large CW has always to endure a long access delay even though currently there may not be any other high priority traffic. Finally, as EDCA supports a relative service differentiation, not an absolute service differentiation, when many low priority traffics contend for the channel, the performance of the high priority traffic will be affected, and its QoS cannot be guaranteed. All these problems are caused by the non-cooperative behaviors of nodes and unfair service differentiation simultaneously, so we call them problem of non-cooperative service differentiation. Due to this problem, the system performance and QoS of both high and low priority traffics will be deteriorated sharply. To our best knowledge, no one has considered this problem.

Game theory is a powerful tool to study situations of conflict and cooperation [8], and recently is widely used for performance analysis and optimization of WLANs. Some researchers [9–12] proposed game models to interpret CSMA/CA and to resolve respectively the selfish problem, the bandwidth allocation problem, and the unfairness problem, which would help to solve the problem of non-cooperative service differentiation. However, the above distributed algorithms require some explicit cooperation among nodes, which is clearly impractical in WLANs. We [13] presented a novel concept of incompletely cooperative game theory to achieve the maximum throughput under different network loads without any explicit cooperation among nodes. Based on the proposed game, a modified EDCA protocol was very briefly presented as a letter [14] to solve the problem of selfish traffic with rational nodes. Its basic assumption is that nodes are cooperative a priori, and would much like to cooperate with each other to solve the problem. And its basic idea is that each node adjusts its contention strategy, e.g., the CW, based on the estimated game state, e.g., the number of competing nodes of each QoS priority.

In this paper, firstly, an incompletely cooperative EDCA protocol (IC-EDCA) will be elaborated in much greater detail, and the preliminary results presented in the letter will be revised and substantially extended. Secondly, to implement IC-EDCA, a frame-analytic estimation algorithm is provided to estimate the number of competing nodes of each QoS priority. Thirdly, a simple and accurate analytical model is postulated, which takes into account the details of IC-EDCA. The results can be used to calculate the saturation channel utility, bandwidth, delay, jitter, and packet-loss-rate for each traffic priority. Finally, extensive performance simulations of the DCF, EDCA, incompletely cooperative game, and IC-EDCA protocols are undertaken.

The rest of this paper is organized as follows. DCF and EDCA are introduced respectively in Section 2. In Section 3, all the aspects of IC-EDCA are discussed in detail. In Section 4, an analytical model is presented to interpret saturation performance of IC-EDCA. In Section 5, simulation studies are carried out to evaluate the performance of IC-EDCA. The concluding remarks are given in Section 6.

2. PRELIMINARY

2.1. Description of DCF

DCF uses a basic acknowledgment mechanism for verifying successful transmissions and an optional RTS/CTS handshaking mechanism for decreasing overhead from collisions. In both cases a binary exponential backoff mechanism is used [15]. If a node has a new packet to transmit, it will monitor the channel activities first. If the channel has been idle for a period of time, i.e., a distributed inter-frame space (DIFS), the node will transmit. Otherwise, if the channel is sensed busy, the node will continuously monitor the channel until it is found to be idle for a DIFS. At this point, the node generates a random backoff interval before transmitting. The backoff time is slotted and the number of backoff slots is uniformly chosen in the range [0,CW−1]. At the first transmission attempt, the CW is set equal to a value CWmin called the minimum contention window. After each unsuccessful transmission, CW is doubled up to the maximum value CWmax = 2m(CWmin + 1)−1. The value m is called the maximum backoff stage, and CWmax is called the maximum contention window. Once CW reaches CWmax, it will remain at the value until the packet is transmitted successfully or the retransmission time reaches the retry limit (r). While the limit is being reached, retransmission attempts will cease and the packet will be discarded.
2.2. Description of EDCA

EDCA is one of the main parts of the IEEE 802.11e standard for service differentiation [16]. In EDCA traffic flows are categorized into four access categories (ACs). Different ACs adopt different values of arbitrary interframe space (AIFS), \( CW_{\text{min}} \), \( CW_{\text{max}} \), and transmission opportunity (TXOP) for acquiring channel access. For example, before transmitting, each node needs to wait for a period of time called AIFS for the channel to be idle. The value of AIFS is associated with the corresponding AC. TXOP is the time interval permitted for a particular node to transmit packets. During the TXOP, there can be a series of frames transmitted by a node. Typically, a shorter AIFS and a longer TXOP is associated with an AC with a higher priority. To introduce better differentiation performance, EDCA extends the basic DCF to support up to four EDCA queues in one QoS-enhanced node. Each of these queues contends to send its packets. However, this paper will not consider the internal collision which may result from this queuing.

Although different ACs adopt different parameter values, these default values of each AC are fixed, and the nodes cannot adjust these contention parameters to the traffic loads.

3. DESCRIPTION OF IC-EDCA

In IC-EDCA, a contention process starts when a new packet arrives at the node’s transmission buffer and ends when the packet is moved out of the buffer (i.e., transmitted successfully or discarded). A contention process may include many time slots. In a time slot, each node estimates the current system state according to its history. After estimating the system state, a node adjusts its own contention parameters, and then takes its actions to contend for the channel. As all the nodes take actions simultaneously, i.e., transmitting their packets or backoff, the node does not know which action the other nodes (i.e., its opponents) are actually taking. However, the node can predict its opponents’ actions according to its history, and make the optimal decisions. In a word, based on the estimated system state, all the nodes cooperate with each other in contending for the channel to solve the problem of non-cooperative service differentiation. As IC-EDCA is based on the estimated system state without any explicit cooperation between nodes, and normally, nodes can obtain only incomplete information (not all the information) of the system state, it is called incomplete cooperation.

The system state includes not only a node’s local states, but also its opponents’ states; the former can be retrieved from DCF or EDCA operation and the latter cannot. Our initial research results [14] showed that the problem of non-cooperative service differentiation is closely related with the number of competing nodes of each AC, i.e., the number of nodes of four ACs that are simultaneously trying to send a packet on the shared medium. For simplicity, the following will only discuss how a node estimates the number of competing nodes of each AC.

3.1. Frame-analytic estimation algorithm

In the infrastructure mode, an access point (AP) can provide the number of active connections by broadcasting an announcement traffic indication message (ATIM) periodically to all the terminals in its cell. However, firstly, the AP cannot provide the number of active connections of each AC; secondly, this involves some delay and in a fast-changing WLAN, more frequent information updates are needed. This may result in a significant overhead. Moreover, this approach is unsuitable for the ad hoc mode, where each terminal has to take a distributed estimation approach.

Bianchi [17] showed that the number of competing nodes can be expressed as function of the collision probability encountered on the channel. Based on this rule, Bianchi presented a Kalman filter estimation of the number of competing nodes. However, as the collision may occur between different ACs, based on the above rule, we cannot estimate the number of competing nodes of each AC. So we presented a frame-analytic estimation algorithm [18], which is easy to be implemented in current WLAN nodes.

During the operation of IEEE 802.11e, as all the frames are broadcasted from a sender node to its destination nodes, the other nodes can obtain some pieces of information from the received frames. For example, IEEE 802.11e provides the MAC address of the sender in the source address (SA) field and the type of AC in the QoS control (QC) field of the MAC header of the transmitted frames. As traffic flows with different MAC addresses and ACs have different values of SAs and QCs, by counting the number of the corresponding frames with different SAs or QCs, the nodes can estimate the number of competing nodes of AC\(_i\) (\(n_i\)), where \(i\) denotes a given AC (0 \(\leq i \leq 3\)). The estimation process includes the following five main steps:

1. To define a variable \(n_i\) to count the number of competing nodes of AC\(_i\).
2. To define an array state\([J][K]\) to record the SAs and QCs of the received frames, and its relevant arrival time values, e.g., state\([1][1]\) records a MAC address (i.e., the value of SA), state\([1][2]\) records its AC (i.e., the value of QC), state\([1][3]\) records the frame arrival time from the SA and QC.
3. When a node detects a new frame, it will compare the SA and QC value of the frame with the recorded SAs and QCs in state\([j][1]\) and state\([j][2]\) (1 \(\leq j \leq J\)). If this is a new SA or QC, the node will increase \(n_i\) by one, and record the SA and QC value and its arrival time in the array state\([J+1][1]\), state\([J+1][2]\), and state\([J+1][3]\) respectively. Otherwise, it will only record the frame arrival time in the array state \([j][K+1]\), where the SA and QC values have been saved in state\([j][1]\) and state\([j][2]\), and the last frame arrival time in state\([j][K]\).
4. To determine if an SA or QC in the array is out of date, that is, how to set the value of the SA or QC survival time (LifeTime). With a smaller LifeTime, the item in the array will be deleted earlier; but with
a larger LifeTime, the node will be less sensitive to the changes. Suppose a node has a frame to send, if the transmission fails, it should get another chance after an average interval called $\Delta t$ to transmit the same frame again; if the transmission fails again, then the node should transmit again after another $\Delta t$ time until it reaches the maximum retry limit $r$. So for simplicity, the survival time for the SA of state[j][1] and the QC of state[j][2] is set as:

$$LifeTime = r \times \Delta t = r \times \sum_{k=3}^{K} \text{state}[j][k] / (K - 2)$$

where the arrays from state[j][3] to state[j][K] record the latest K-2 frame arrival intervals from the traffic flow with the SA of state[j][1] and the QC of state[j][2]. Corresponding to an SA, if no frame is transmitted from this node when its LifeTime has elapsed, it is assumed that this node has left the WLAN. Corresponding to a QC, if no frame is transmitted from this node when its LifeTime has elapsed, it is assumed that this traffic flow (i.e., AC) has been closed. Moreover, the estimated LifeTime needs to be updated periodically. However, discussions in detail of the run-time estimation methods are beyond the scope of this paper.

(5) To scan the whole array. If a SA or QC is found to be out of date, the node will delete this item and decrease $n_i$ by one.

### 3.2. Adaptive contention algorithm

Research results [13] show that the system performance is closely related with the CW min, and the optimal value of CW min is dependent on the number of competing nodes. After estimating the number of competing nodes of AC i ($n_i$), each node tunes the value of its CW min as follows:

$$CW_{i,\text{min}} = \lfloor n_i \times \text{rand}(7, 8) \rfloor \quad 0 \leq i \leq 3$$  \hspace{1cm} (1)

where $\text{rand}(x, y)$ returns a random value between $x$ and $y$, $\lfloor z \rfloor$ is the largest integer that is not more than $z$. We [14] showed that the above equation provides the optimal value of CW min when considering only one AC in a WLAN cell, and the optimal CW min corresponds to the smallest collision probability.

According to IEEE 802.11e, if $j$ ($0 \leq j \leq r$) denotes the number of successive failed attempts to transmit (due to collision), the current value of CW variable of AC i can be obtained as:

$$CW_{i,j} = \begin{cases} 
CW_{i,\text{min}} & j = 0 \\
\min(2CW_{i,j-1}, CW_{i,\text{max}}) & \text{else}
\end{cases}$$  \hspace{1cm} (2)

where $\min(x, y)$ returns the smaller one of $x$ and $y$. The CW max of AC i, $CW_{i,\text{max}}$, is defined in IEEE 802.11e.

However, if considering multiple ACs simultaneously, not only CW min but also the backoff strategy should be adjusted to the estimated system state. Then the current backoff time is defined as:

$$\text{Backoff}_{i,j} = \text{rand}(W_{i,j,\text{low}}, W_{i,j,\text{up}})$$  \hspace{1cm} (3)

where the lower limit

$$W_{i,j,\text{low}} = \sum_{k=3}^{3} CW_{k,j} = \begin{cases} 
0 & i = 3 \\
\sum_{k=3}^{i} CW_{k,j} & 0 \leq i \leq 2
\end{cases}$$  \hspace{1cm} (4)

and the upper limit

$$W_{i,j,\text{up}} = \sum_{k=3}^{3} CW_{k,j}$$  \hspace{1cm} (5)

Hence, at the first transmission attempt, nodes of AC 3, AC 2, AC 1, and AC 0 use the window $[0, CW_{3,\text{min}}]$, $[CW_{5,\text{min}}, CW_{3,\text{min}} + CW_{2,\text{min}}]$, $[CW_{3,\text{min}} + CW_{2,\text{min}}, CW_{3,\text{min}} + CW_{2,\text{min}} + CW_{1,\text{min}}]$, and $[CW_{3,\text{min}} + CW_{2,\text{min}} + CW_{1,\text{min}}, CW_{3,\text{min}} + CW_{2,\text{min}} + CW_{1,\text{min}} + CW_{0,\text{max}}]$ respectively. After each unsuccessful transmission, the windows are doubled up to the maximum values, i.e., $CW_{3,\text{max}}$, $CW_{3,\text{max}} + CW_{2,\text{max}}$, $CW_{3,\text{max}} + CW_{2,\text{max}} + CW_{1,\text{max}}$, and $CW_{3,\text{max}} + CW_{2,\text{max}} + CW_{1,\text{max}} + CW_{0,\text{max}}$.

After estimating the system state, all the nodes cooperate with each other to contend for the channel, i.e., adjusting their CWs to the estimated number of competing nodes of each AC. Given that there are many higher priority traffics (e.g., AC i), firstly, the best strategy for the lower priority traffic (e.g., AC 1) is to move its CW forward behind the higher priority traffic in order to defer its transmission after the higher priority traffic. In this way, the transmission of the lower priority traffic is prevented from disturbing the transmission of the higher priority traffic. Secondly, the high priority traffics would adjust their CWs to the estimated system state, in order to decrease the collisions among themselves. So IC-EDCA can reduce collisions, and ensure the QoS of the high priority traffic. Given that there is not any higher priority traffic, the best strategy for the lower priority traffic is to move its CW forward in order to advance its transmission. In this way, IC-EDCA can make effective use of the channel. Please note that IC-EDCA does not mess up the service differentiation in IEEE 802.11e as the low priority traffic will get a high priority to access the channel only when there is not any higher priority traffic.

### 4. PERFORMANCE ANALYSIS

The third contribution of this paper is the analytical evaluation of the saturated channel utility, bandwidth, delay, jitter,
and packet-loss-rate of each AC under the assumption of ideal channel conditions (e.g., not considering hidden terminals). In saturation conditions, each node has a packet available for transmission immediately after the completion of each successful transmission [19]. The performance of IC-EDCA can be best observed under saturated conditions. Under unsaturated conditions some problems may not be visible. Therefore, the analytical model for saturation performance is discussed below.

The proposed model is based on the Bianchi’s model, bi-dimensional Markov chain, for DCF [19]. There are quite a few papers on the analysis of EDCA and we may look at those analytical models rather than using the Bianchi’s model. However, as the internal collision is not considered in this paper, we can model each traffic class accordingly just like DCF works. So, for simplicity, the analytical model for IC-EDCA is obtained by modifying the Bianchi’s model.

### 4.1. Tri-dimensional Markov chain model

Assume each node belongs to only one AC, i.e., as stated internal collision is not being considered. Let us consider the contention procedure of a given node of AC. Let \( s(t) \) be the stochastic process representing the backoff stage of the AC node at time \( t; s(t) \in [0, r] \). Let \( b(t) \) be the stochastic process representing the backoff time counter of the AC node at time \( t; b(t) \in [W_{i,j,low}, W_{i,j,up} - 1] \). A tri-dimensional Markov chain \( \{i, s(t), b(t)\} \) is shown in Figure 1 under the assumption that \( p_i \), the probability that a transmitted packet of AC_i collides, is independent of the backoff procedure. Let the stationary distribution of the chain be

\[
b_{i,j,k} = \lim_{t \to \infty} P[i, j = s(t), k = b(t)]
\]

\[
0 \leq i \leq 3, 0 \leq j < r, W_{i,j,low} \leq k \leq W_{i,j,up} - 1
\]

By analyzing the chain, the following equations are obtained:

\[
b_{i,j-1,0} \cdot p_i = b_{i,j,0} \Rightarrow b_{i,j,0} = (p_i)^3 b_{i,0,0} \quad 1 \leq j < r
\]

\[
b_{i,r-1,0} \cdot (1 - p_i) = b_{i,r,0} = \frac{p_i}{1 - p_i} b_{i,r-1,0}
\]

\[
= \frac{(p_i)^3}{1 - p_i} b_{i,0,0}
\]

\[
b_{i,j,k} = \begin{cases} \frac{W_{i,j,up} - k}{W_{i,j,up} - W_{i,j,low}} b_{i,j,0} & W_{i,j,low} \leq k \leq W_{i,j,up} - 1 \\ 0 & k < W_{i,j,low} \end{cases}
\]

Hence, all the values of \( b_{i,j,k} \) can be expressed as functions of \( b_{i,0,0} \) and the conditional collision probability \( p_i \). By substituting the above equations into the normalization condition of the chain, the following equation is obtained:

\[
1 = \sum_{j=0}^{r} \sum_{k=0}^{r} b_{i,j,k}
\]

\[
= \sum_{j=0}^{r} \sum_{k=0}^{r} b_{i,j,0} + \sum_{j=0}^{r} \sum_{k=W_{i,j,low}}^{W_{i,j,up} - 1} \frac{W_{i,j,up} - k}{W_{i,j,up} - W_{i,j,low}} b_{i,j,0}
\]

\[
= \sum_{j=0}^{r} W_{i,j,low} b_{i,j,0} + \sum_{j=0}^{r} \frac{W_{i,j,up} - W_{i,j,low} + 1}{2} b_{i,j,0}
\]

\[
= \sum_{j=0}^{r} \frac{W_{i,j,up} + W_{i,j,low} + 1}{2} b_{i,j,0}
\]

If supposing \( m_i < r \), based on Equations (4) and (5), the above equation can be rewritten as

\[
1 = \frac{1}{2} \left( \sum_{j=0}^{m_i} 2^j Ab_{i,j,0} + \sum_{j=m_i+1}^{r-1} 2^j Ab_{i,j,0} + 2^m Ab_{i,r,0} \right)
\]

\[
+ \sum_{j=0}^{r-1} b_{i,j,0} + b_{i,r,0}
\]

\[
= \frac{b_{i,0,0}}{2} \left( 1 - (2p_i)^{m_i} \right) A + (p_i)^{m_i+1} \frac{2^{m_i} A}{1 - p_i}
\]

where

\[
A = \sum_{k=1}^{3} W_{k,\min} + \sum_{k=r+1}^{3} W_{k,\min}
\]
Hence,
\[
b_{0,0,0} = \frac{2(1 - 2p_i)(1 - p_i)}{A(1 - (2p_i)^{n_i + 1})(1 - p_i) + (1 + 2^{n_i} A(p_i)^{n_i + 1})(1 - 2p_i)} \quad r > m_i
\]
(13)

If supposing \( m_i > r \), in the similar way, the following equation is obtained:
\[
b_{0,0,0} = \frac{2(1 - 2p_i)(1 - p_i)}{A(1 - (2p_i)^{n_i + 1})(1 - p_i) + (1 + 2^{n_i} A(p_i)^{n_i + 1})(1 - 2p_i)} \quad r \leq m_i
\]
(14)

Then the probability that the AC_i node transmits packets in a randomly chosen time slot can be expressed as
\[
\tau_i = \sum_{j=0}^{r} b_{i,j,0} = 1 - (p_i)^{r+1} \prod_{h=0}^{m_i} (1 - \tau_h)^{s_h}
\]
(15)

And its collision probability is
\[
P_{u,i} = 1 - (1 - \tau_i)^{n_i}
\]
(16)

By solving the above two equations, the values of \( p_i \) and \( \tau_i \) can be found.

Consider a fixed number \( n \) of contending nodes, and \( n = \sum n_i \). For simplicity, assume the number of contending nodes is estimated accurately in this analytical model. The probability that at least one AC_i node transmits in a given time slot can be derived as follows:

\[
P_{u,i} = 1 - (1 - \tau_i)^{n_i}
\]
(17)

The probability that exactly one AC_j node will transmit packets on the channel, assuming that at least one AC_i node will transmit, can be expressed as

\[
P_{s,i} = \frac{n_i \tau_i (1 - \tau_i)^{n_i - 1} \prod_{h=0}^{m_i} (1 - \tau_h)^{s_h}}{1 - (1 - \tau_i)^{n_i}}
\]
(18)

As regards to the status of packet transmission in a given time slot, there are three possibilities. Firstly, there may be no transmission with a probability of \( P_i = \prod_{i=0}^{m_i} (1 - \tau_i)^{s_i} \), whose time is \( \sigma \). \( \sigma = \text{ASlotTime}, \) which is defined in IEEE 802.11x. Secondly, it contains a successful transmission. The probability that an AC_i node transmits successfully is \( P_{u,i} P_{s,i} \), whose average time is \( T_{s,i} \). In the basic acknowledgement mechanism, \( T_{s,i} = F_i + \text{SIFS + ACK + AIFS}[i] + \delta \), where \( F_i \) is the average time length of the AC_i data-frame, SIFS, short interframe space, is defined in IEEE 802.11x; \( \delta \) is the propagation delay, ACK is the time length of the acknowledgement frame, and AIFS[i] is the AIFS value of AC_i. Finally, there may be collision with a probability of \( P_i = 1 - P_e - \sum_{i=0}^{3} P_{u,i} P_{s,i} \), whose average time is \( T_{c,i} \). In the basic acknowledgement mechanism, \( T_{c,i} = F_i + \text{AIFS}[i] + \delta \), where \( F_i \) is the average length of the longest frame involved in the collision, was discussed in detail in Reference [19]. So the average length of time slots is obtained

\[
\bar{S} = P_e \sigma + \sum_{i=0}^{3} P_{u,i} P_{s,i} T_{s,i} + P_c T_{c,i}
\]
(19)

Let the average time length of payload of the AC_i data-frame be \( T_p \). Thus, the channel utility \( \eta \) can be expressed as

\[
\eta = \frac{\text{Average payload successfully transmitted in a slot}}{\text{Average length of a slot}} = \frac{\sum_{i=0}^{3} P_{u,i} P_{s,i} T_{s,i}}{P_e \sigma + \sum_{i=0}^{3} P_{u,i} P_{s,i} T_{s,i} + P_c T_{c,i}}
\]
(20)

### 4.2. Bandwidth, delay, jitter, and packet-loss-rate

Let the average bit size of payload in the AC_i data-frame be \( E_i \). Then the average bandwidth of an AC_i node (i.e., node throughput) is

\[
B_i = \frac{P_{u,i} P_{s,i} E_i}{P_e \sigma + \sum_{i=0}^{3} P_{u,i} P_{s,i} T_{s,i} + P_c T_{c,i}}
\]
(21)

According to Figure 2, the probability that an AC_i node successfully transmits a packet at the \( j \)th backoff stage is

\[
P_{s,i} = (1 - p_j)(p_j)^j \quad 0 \leq j \leq r
\]
(22)

And its average delay is

\[
T_{i,j} = \text{AIFS}[i] + \delta \sum_{k=0}^{j} \frac{W_{k,up} + W_{k,low}}{2} + j \cdot T_c
\]
(23)

In fact, during the backoff process, other nodes may be allowed to transmit their frames. That is to say, the back-
off process may be interrupted. Hence, the above equation should be changed to

$$T_{i,j} = AIFS[i] + S \cdot \sum_{k=0}^{j} \frac{W_{k,up} + W_{k,low}}{2} + j \cdot T_c \quad 0 \leq j \leq r$$  \hspace{1cm} (24)

Hence, the average access delay of an AC node is

$$D_i = \sum_{j=0}^{r} P_{i,j} T_{i,j}$$  \hspace{1cm} (25)

As shown in Figure 1, the current (not average) access delay that a packet of the AC node is transmitted successfully at the $j$th backoff stage is

$$X_{i,j,k} = AIFS[i] + k \cdot S + j \cdot T_c$$  \hspace{1cm} (26)

where $k$ is the sum of backoff slots that the packet encounters, and $K_{i,j} = \sum_{k=0}^{j} (W_i - 1)$.

Let the probability that a packet of the AC node is transmitted successfully at the $j$th backoff stage and encounters $k$ backoff slots be $P_{i,j,k}$. The jitter (i.e., delay variation) of the AC node can be expressed as

$$J_i = E \left( \left| X_{i,j,k} - E \left( X_{i,j,k} \right) \right| \right) = \sum_{j=1}^{r} \sum_{k=0}^{K_{i,j}} P_{i,j,k} \cdot \left| X_{i,j,k} - D_i \right|$$  \hspace{1cm} (27)

However, it is too complex to calculate the probability $P_{i,j,k}$. Hence, an approximation of $J_i$ is given in this paper.

$$J_i = E \left( \left| T_{i,j} - E \left( T_{i,j} \right) \right| \right) = \sum_{j=1}^{r} P_{i,j} \cdot \left| T_{i,j} - D_i \right|$$  \hspace{1cm} (28)

The packet-loss-rate for the AC node is

$$P_{i,d} = \left( p_i \right)^{r+1}$$  \hspace{1cm} (29)

### 5. SIMULATION RESULTS

To evaluate the performance of IC-EDCA, and the accuracy of the proposed analytical model, extensive simulations are carried out with NS-2 in the following. The values of the parameters used to obtain numerical results for both the analytical models and simulations are specified in IEEE 802.11x. The WLAN works in the ad hoc mode, i.e., there is a single collision domain and all the nodes compete for the same channel. The channel is assumed to be ideal without hidden terminals and capture. The channel data rate is fixed at 11 Mbps. Performance of IC-EDCA, incompletely cooperative game, EDCA, and DCF are compared under the same conditions. Assume that the number of nodes increases from 5 to 30 in a step of 5, and that each node has a packet available for transmission immediately after the transmission buffer is empty. As the proposed IC-EDCA is generic, all the four levels of ACs can be designated. Yet for simplicity, it is assumed that each node belongs to one AC, and traffic flows are classified into two ACs with equal number of nodes (i.e., $n_3 = n_2$, $n_1 = n_0 = 0$). For simplicity, the frame will be discarded only as the retransmission time reaches the retry limit. The payload sizes for both ACs are fixed at 1023 bytes.

Figures 2–6 show that the analytical results are close to simulation results, which implies that the proposed analytical models are valid. The analytical model is based on the assumption that all nodes know the number of competing nodes of each AC, and the simulation results are achieved with the frame-analytic estimation algorithm. So Figures 2–6 also show that the frame-analytic estimation algorithm is accurate.

Figure 2 shows that the channel utility of IC-EDCA and incompletely cooperative game is larger than that of DCF, which is much larger than that of EDCA. The channel util-

![Figure 2. Saturation channel utility.](image)

![Figure 3. Saturation bandwidth of an AC node.](image)
Service differentiation in WLANs

L. Zhao et al.

Figure 4. Saturation access delay of an AC node.

Figure 5. Saturation jitter of an AC node.

Figure 6. Saturation packet-loss-rate of an AC node.

The channel utility of the four protocols becomes lower with the increase in the number of competing nodes as collisions increase. However, the channel utility of EDCA decreases very sharply. In EDCA, to support service differentiation, compared with DCF, the AC3 and AC2 traffic has a smaller CW, which causes more collisions when the number of the two ACs nodes gets large, i.e., the problem of non-cooperative service differentiation. The more collisions there are, the more sharply the channel utility decreases. Incompletely cooperative game aims at maximizing the channel utility under different network loads, and its channel utility almost keeps constant. In IC-EDCA, each node adjusts its CW adaptively to the changing traffic loads (i.e., the number of each AC nodes), so the channel utility of IC-EDCA is very close to that of incompletely cooperative game.

Figure 3 shows that, in EDCA the average bandwidth of an AC3 node decreases more sharply than that of an AC2 node, and the two bandwidths get closer with the increasing of the number of competing nodes. It indicates EDCA supports relative service differentiation. In IC-EDCA, the average bandwidth of an AC3 node is much larger than that of an AC2 node, and the two values keep almost constant. It indicates IC-EDCA supports absolute service differentiation. In incompletely cooperative game and DCF, the two AC nodes have the same bandwidth, which indicates the two protocols do not support any service differentiation. Obviously, EDCA supports service differentiation, and the game supports cooperation; however, no one can solve the problem of non-cooperative service differentiation.

To evaluate the analytical model, a new packet is generated only after the current packet is transmitted successfully or discarded. In this case, the average access delay of an AC node in IC-EDCA is larger than that in EDCA, as shown in Figure 4. The reason is that much more packets are discarded in EDCA as shown in Figure 6.

Due to the approximation of $J_i$ in Equation (28), the analytical results of average jitter of an AC node are not so accurate as the other parameters, as shown in Figure 5. Figure 6 shows that the average packet-loss-rate of an AC node in EDCA increases very sharply due to the problem of non-cooperative service differentiation. In IC-EDCA, each node can adjust its contention strategy to the current system state, and transmit its packets successfully before the retransmission time reaches the retry limit. So the average packet-loss-rate of an AC node in IC-EDCA is always zero.

In saturation conditions, after the completion of each successful transmission, each node immediately has a packet available for transmission. In fact, the load of the stations is not fixed. Some nodes may have data to send during that time, and some others may not. To simulate the performance of IC-EDCA in the real environments, the following simulations in unsaturated conditions with hidden terminals and capture are carried out, where a two-ray ground reflection model is used. We consider a rectangular area of $500 \times 500$ m and 40 nodes are randomly distributed in this area. These nodes are classified into four ACs with equal number of nodes (i.e., $n_1 = n_2 = n_3 = n_0 = 10$), and each node generates new packets under a Poisson process. The packet arrival rate is initially set lower than the saturation case, and then subsequently increased so that, at the end of the simulation time, all nodes are in a saturation condition.

When the network load is light, Figures 7–9 show that the performance of DCF, EDCA, and IC-EDCA are similar.
When the load is heavy, they have distinct performances in terms of system throughput, bandwidth, and access delay. IC-EDCA works best, and due to the problem of non-cooperative service differentiation, EDCA performs even worse than DCF.

As shown in Figure 7, at the beginning of the simulation, the system throughput of the three protocols increases with the increase in the network load, while at the end of the simulation it gets saturated at 6 Mb/s in IC-EDCA and 5 Mb/s in DCF, and drops sharply to about 2 Mb/s in EDCA. Moreover, Figures 7–9 show that EDCA, DCF and IC-EDCA get saturated in turn with the increasing of the network load.

Compared with DCF, to support service differentiation, the AC3 and AC2 traffics in EDCA have a much smaller CW. It causes the problem of non-cooperative service differentiation, i.e., more collisions when the number of competing AC3 and AC2 nodes increases with the increasing of the simulation time. So EDCA gets saturated before DCF. Due to the problem, the system throughput in EDCA is much lower than that in DCF and decreases very sharply in the overload condition. In EDCA, although the AC3 and AC2 traffics have a higher priority than the AC1 and AC0 traffic, they have a lower bandwidth and a larger delay. In IC-EDCA, the traffic with a higher priority always has a better QoS metrics than the traffic with a lower priority.

Moreover, Figure 7 shows that the system throughput of IC-EDCA with ideal and actual estimation are close to each other, which indicates that the frame-analytic estimation algorithm is also accurate enough after considering hidden terminals and capture.

In general, in IC-EDCA all the nodes cooperate with each other, i.e., adjusting their CWs adaptively to the estimated number of competing nodes of each AC, so IC-EDCA can solve the problem of non-cooperative service differentiation, and performs much better than EDCA in terms of system throughput and QoS.

6. CONCLUSION AND FUTURE WORKS

In this paper, IC-EDCA, an incompletely cooperative EDCA protocol is used in WLANs to solve the problem of non-cooperative service differentiation. To our best knowledge, no one has considered this problem. In IC-EDCA, to implement the optimal contention strategy, all the nodes adjust their contention parameters, e.g., the CWs, adaptively to the estimated system state, e.g., the number of competing nodes of each AC. A frame-analytical estimation algorithm is presented to implement IC-EDCA in current WLAN nodes. Moreover, an analytical model is proposed to analyze the performance of IC-EDCA in terms of channel utility, delay, jitter, and packet-loss-rate. The simulation results show that IC-EDCA is an appropriate tool to improve throughput, and to decrease delay, jitter, and packet-loss-rate under overload conditions, and that the proposed analytical model is valid.

Currently we are carrying research further on the following two fields. Firstly, we are improving the frame-analytic estimation algorithm as it may not be perfectly correct under some scenarios. In these cases, each node should establish its own belief on the system parameters so that its strategy
can be adjusted accordingly. Secondly, we are improving the analytical model to consider the internal collision.

ACKNOWLEDGEMENTS

This work is supported by the 111 Project (B08038), State Key Laboratory of Integrated Services Networks (ISN090105), Program for New Century Excellent Talents in University (NCET-08-0810), National Natural Science Foundation of China (No. 60772137), and UK-China Science Bridges: R&D on (B)4G Wireless Mobile Communications.

REFERENCES

4. Engelstad PE, Osterbo ON. Analysis of the total delay of IEEE 802.11e EDCA and 802.11 DCF. IEEE International Conference on Communications (ICC), 2006, Turkey, 552-559.
5. Tao Z, Panwar S. Throughput and delay analysis for the IEEE 802.11e enhanced distributed channel access. IEEE Transactions on Communications 2006; 54(4): 596-603.

AUTHORS’ BIOGRAPHIES

Dr Liqiang Zhao is a Professor of Communications and Information Systems within State Key Laboratory of Integrated Services Networks, Xidian University, China. His current research focuses on broadband wireless access and green communications. Moreover, he is researching inter-disciplinary topics, e.g., introducing game theory from mathematics and economics into broadband wireless access and green networks. Since 2005, he has been awarded more than eight projects funded by a number of sources, e.g., Europe, Chinese government, and direct industrial funding. His participation in various projects has yielded a number of concrete results including enormous high-level publications, etc. Due to his excellent works in research, he was awarded by the Program for New Century Excellent Talents in University, Ministry of Education, China in 2008.

Li Cong received her B.Sc. degree in Communications Engineering from Xidian University, Xi’an, Shaanxi, China in 2006, and she is working towards her Ph.D. in Communications and Information Systems at Xidian University. She is currently working on dynamic resource management problem by introducing game theory from...
mathematics and economics. And her research interests include cooperative transmission, spectrum and power management in wireless Ad Hoc and Sensor networks.

Wei Ding obtained his B.Sc. degree in Automatic Control from Beijing Information Science and Technology University, China in 2002, and his M.Sc. degree in Internet Computing from University of Bedfordshire, UK in 2005, and then joined the Institute for Research in Applicable Computing, University of Bedfordshire in 2006 to start his Ph.D. career. His current research interest is 3G/4G network Planning and Optimization.

Dr Xinhe Wang is a senior lecturer in wireless communications and head of Intelligent Wireless Networks for Healthcare (IWinHealth) research group in School of Engineering, Swansea University. His current research interests include wireless mesh and sensor networks, cognitive radio networks, and software defined radios and applications of wireless networks in healthcare.

Dr Hailin Zhang is a Professor in Telecommunications and Dean of School of Telecommunications Engineering, Xidian University, China. His current research interests include multimedia communications, wireless high data rate digital transmission, etc. He has published over 50 papers in refereed journals and conferences, and been awarded many projects by the research councils and the telecom industry.