[1] Analysis of the IEEE 802.11 DCF with Service Differentiation Support in Non-Saturation Conditions

Bo Li and Roberto Battiti

Department of Computer Science and Telecommunications, University of Trento,
38050 POVO Trento, Italy
{li,battiti}@dit.unitn.it

Abstract. Although the performance analysis of the IEEE 802.11 Distributed Coordination Function (DCF) in saturation state has been extensively studied in the literature, little work is present on performance analysis in non-saturation state. In this paper, a simple model is proposed to analyze the performance of IEEE 802.11 DCF with service differentiation support in non-saturation states, which helps to obtain a deeper insight into the IEEE 802.11 DCF. Based on the proposed model, we can approximately evaluate the most important system performance measures, such as packet delays, which provide one with an important tool to predict and optimize the system performance. Moreover, a practical method to meet packet delay requirements is presented based on our theoretical results. Comparisons with simulations show that this method achieves the specified packet delay requirements with good accuracy.

Keyword: Wireless LAN, IEEE 802.11, Quality of Service Guarantee, Service Differentiation

1 Introduction

In recent years, IEEE 802.11 has become one of the most important international standards for Wireless Local Area Networks (WLAN's) [1]. In the IEEE 802.11 protocol, the fundamental mechanism to access the medium is the Distributed Coordination Function (DCF), which is a random access scheme based on the carrier sense multiple access with collision avoidance (CSMA/CA) protocol. Many performance analyses of 802.11 have been proposed, such as those in [2]-[5]. However, the previous papers consider the assumption of saturation state. That is, it is assumed that the transmission queue for each station is always nonempty, which is not realistic in real-world systems. In [6] and [7], more practical queuing models for IEEE 802.11 DCF are proposed which incorporate practical packet arrival processes. However, the service rate for each node is still based on the results obtained in [4],

1. This work is supported by the project of WILMA funded by Provincia Autonoma di Trento (www.wilmaproject.org)

where saturation state is assumed. The limitation is overcome in [8], where performance analysis in non-saturation state is considered by introducing probability generating functions, which allow the computation of the probability distribution function (pdf) of the delay. However, computing pdf values with the proposed method has a high computational cost and therefore the approach is of limited practical use. The other drawback is that the complex analysis method in [8] is of little help to obtain deeper insight into relationships among different system parameters. Moreover, service differentiation support is not considered. In this paper, based on our former work in [9]-[10], a simple analysis model is proposed to analyze the performance of an enhanced 802.11 DCF with service differentiation support in non-saturation state. We considered the following objectives when defining the model.

1. The analysis model should be simple enough to obtain a clear insight into relationships among the most important system parameters.
2. The analysis model should be as practical as possible, so that it can be implemented in real-world systems.
3. Service differentiation must be considered.

2 Performance Analysis

We consider a single-hop wireless LAN, where stations can "hear" each other well. It is assumed that the channel conditions are ideal (i.e., no hidden terminals and capture). M types of traffic are considered with n_i type i $(i = 1, ..., M)$ stations, and, for simplicity, each station bears only one traffic flow. If the station is busy on the arrival of a packet, the packet must wait in the corresponding transmission queue. The buffer size is assumed to be infinite. It is assumed that the packet arrival processes for type i traffic flows follow independent and identical distributions (i.i.d.), with mean

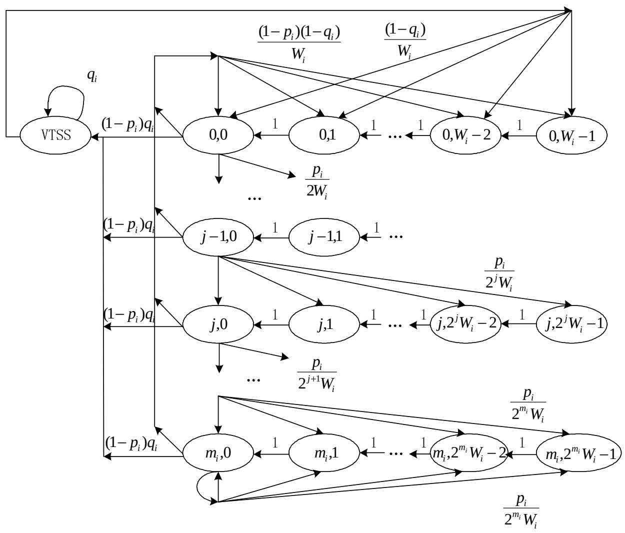

Fig. 1. Markov model for a type *i* traffic flow in ADCF

packet inter-arrival duration $T_{p,i}$. The model can consider different arrival processes. Moreover, it is assumed that all packets have the same payload length, which is transmitted in the duration of P_L . It is also assumed that a backoff process starts immediately when the current packet arrives at the head of the queue.

In the following, a type i traffic flow is considered. Let $b_i(t)$ be the stochastic process representing its backoff time counter. Moreover, let us define $W_i = CW_{\min,i}$ as its minimum contention window. Denote m_i , "maximum backoff stage" as the value such that $CW_{\max,i} = 2^{m_i} \cdot W_i$. $s_i(t)$ is the stochastic process representing its backoff stage $(0,1,...,m_i)$. A two-dimensional discrete-time Markov chain (shown in Fig. 1) is used to model the behavior of the traffic flow. The states are defined as combinations of two integers $\{s_i(t), b_i(t)\}$. It should be noted that apart from using states $\{s_i(t), b_i(t)\}$, a state VTSS (Virtual Time Slot State) is used to model the case that a traffic flow has finished sending a packet and is waiting for the next one. In order to make the system tractable by using a discrete-event Markov chain, the VTSS is sub-divided into different VTS (Virtual Time Slots), whose duration is the same as the time slot in the backoff process. We assume that the station checks if there is a packet available for transmission only at the end of a VTS. In this way, the behavior of the traffic flow in VTSS can be modeled in the same way as the actual backoff processes. For clarity, the above approximated version of DCF is called ADCF. This approximation has very little influence to the final system performance, as verified by extensive simulations. If it is found that the packet transmission queue is not empty after sending the current packet, the state of the traffic flow transits from VTSS to some backoff state. Otherwise, the traffic flow still needs to wait for the arrival of the next packet in VTSS. From Fig. 1, it can be seen that after a packet has been successfully sent or the current VTS has finished, the traffic flow steps into another VTSS with probability q_i . Moreover, parameter p_i is referred to as conditional collision probability, the probability of a collision seen by a packet belonging to a type i traffic flow at the time of its being transmitted on the channel. For simplicity, both q_i and p_i are regarded as constant, which is validated through extensive simulations.

In steady state, $d_{j,k}(i) \equiv \lim_{t\to\infty} P\{s_i(t) = j, b_i(t) = k\}$ ($i = 1,...,M$, $j \in [0, m_i]$, $k \in [0, 2^j W_i - 1]$) is the stationary distribution of backoff states of a type i traffic flow. $P_{VTSS,i}$ is defined as the probability for the traffic flow being at VTSS. Therefore, based on the Markov chain, we have

$$P_{VTSS,i} = d_{0,0} \cdot q_i / (1 - q_i) \tag{1.1}$$

$$\begin{cases} d_{j,0}(i) = p_i^{\,j} \cdot d_{0,0}(i) & (0 < j < m_i) \\ d_{m_i,0}(i) = p_i^{m_i} \cdot d_{0,0}(i) / (1 - p_i) \end{cases} \tag{1.2}$$

$$d_{j,k}(i) = (2^j W_i - k) \cdot d_{j,0}(i) / 2^j W_i \tag{1.3}$$

τ_i is the probability that a type i traffic flow transmits in a randomly chosen time slot. It can be given as

$$\tau_i = \sum_{j=0}^{m_i} d_{j,0}(i) = d_{0,0}(i)/(1-p_i) \quad (2)$$

Since $P_{VTSS,i} + \sum_{j=0}^{m_i} \sum_{k=0}^{2^j W_i - 1} d_{j,k}(i) = 1$, combing equations 1 and 2, we have

$$\left\{(1-2p_i)(W_i+1) + p_i W_i [1-(2p_i)^{m_i}]\right\} \cdot \tau_i \Big/ [2(1-2p_i)] + P_{VTSS,i} = 1 \quad (3)$$

Extensive simulations show that, even if the packet arrival of each traffic flow are assumed to be independent, in some cases there are obvious correlations between behaviors of different traffic flows. Therefore, by introducing compensation factors $\alpha_i > 0 \ (i = 1,...,M)$, packet collision rates can be expressed as

$$p_i = \alpha_i \cdot [1-(1-\tau_i)^{n_i - 1} \prod_{j=1, j \neq i}^{M} (1-\tau_j)^{n_j}] \quad (4)$$

In non-saturation state, the system total throughput S and throughputs $S_i \ (i = 1,...,M)$ contributed by type i traffic flows can be expressed as follows with the assumption that all the arrived packets are finally transmitted successfully

$$S = \sum_{i=1}^{M} S_i = \sum_{i=1}^{M} \frac{n_i P_L}{T_{p,i}} = \frac{P_L \cdot \sum_{i=1}^{M} n_i \cdot \tau_i \cdot (1-p_i)}{\left(\begin{array}{l} \beta \cdot \sigma \cdot \prod_{i=1}^{M} (1-\tau_i)^{n_i} + P_s \cdot \sum_{i=1}^{M} n_i \cdot \tau_i \cdot (1-p_i) + \\ [1-\beta \cdot \prod_{i=1}^{M} (1-\tau_i)^{n_i} - \sum_{i=1}^{M} n_i \cdot \tau_i \cdot (1-p_i)] \cdot P_c \end{array} \right)} \quad (5)$$

where $\beta > 0$ is another compensation factor. It should be noted that the purpose for the introduction of α_i and β is to make our mathematical expressions more rigorous. Extensive experiments show that α_i and β can be approximated as one under the case that the system operates in stable states. Moreover, in equation 5, σ is the duration of an empty time slot (it is also the duration of an empty VTS). P_s is the average time of a slot because of a successful transmission of a packet. And P_c is the average time the channel is sensed busy by each station during a packet collision. We have:

$$P_s = PHY_{header} + MAC_{header} + P_L + SIFS + \delta + ACK + DIFS + \delta \quad (6)$$

$$P_c = PHY_{header} + MAC_{header} + P_L + DIFS + \delta \quad (7)$$

where δ is the propagation delay.

We assume that behaviors of all the traffic flows are independent (simulations show that this assumption approximately holds in the case that minimum contention window sizes W_i s are not very small). In this case, the above introduced α_i and β can be approximated as 1. Therefore, given the corresponding offered traffic load (hence, the system throughput S is also given), based on equations 4 and 5, packet sending rates τ_i s and the corresponding packet collision rates p_i s can be determined. Although two sets of solutions can be obtained, only one is preferred, which corresponds to smaller packet collision rates. We denote the preferred solution as

$\Phi(\tau^*_{1,...,M}, p^*_{1,...,M})$. Considering the stability, the system should operate close to this solution. It can be seen that $\Phi(\tau^*_{1,...,M}, p^*_{1,...,M})$ can be completely determined without relying on the measurements of other system parameters, such as, packet collision rate p_i s.

Next, we make an analysis on packet delay $T_{d,i}$ $(i = 1,...,M)$, which is defined as the average duration between the beginning of a backoff procedure and the instant that the corresponding packet has been successfully sent. Let us consider a type i traffic flow. $\overline{n}_{VTS,i}$ is the average number of successive VTS following the successful sending of a packet. It can be given as

$$\overline{n}_{VTS,i} = \sum_{j=1}^{\infty} j \cdot q_i{}^j \cdot (1-q_i) = \frac{q_i}{1-q_i} = \frac{P_{VTSS,i}}{d_{0,0}(i)} = \frac{P_{VTSS,i}}{\tau_i(1-p_i)} \quad (8)$$

Considering the case that $P_s \approx P_c$ and $\tau_i \ll 1$, the average duration $T_{VTS,i}$ of a *VTS* can be approximated as

$$T_{VTS,i} \approx \beta \cdot \sigma \cdot \prod_{j=1}^{M} (1-\tau_j)^{n_j} + [1 - \beta \cdot \prod_{j=1}^{M} (1-\tau_j)^{n_j}] \cdot P_s - \tau_i P_s \quad (9)$$

According to equation 5, we have

$$T_{VTS,i} \approx n_i \tau_i (1-p_i) P_L / S_i - \tau_i P_s \quad (10)$$

Therefore, $T_{d,i}$ can be approximated as

$$T_{d,i} = T_{p,i} - \overline{n}_{VTS,i} \cdot T_{VTS,i} \approx \frac{n_i P_L}{S_i}(1 - P_{VTSS,i}) + \frac{P_s}{1-p_i} \cdot P_{VTSS,i} \quad (11)$$

It should be noted that the above estimated $T_{d,i}$ can be approximated as the average service time for a packet in its transmission queue. Therefore, it can be directly applied to evaluate the average packet queuing delay (waiting time in transmission queues) by using G/G/1 queuing model [11], which is omitted here because of space limitation.

3 Approximation Analysis

Assume that the system operation point can be approximated as $\Phi(\tau^*_{1,...,M}, p^*_{1,...,M})$. Theoretical analysis shows that if the number of traffic flows n_i $(i = 1,...,M)$ and the packet payload P_L is not too small, it is reasonable to assume that $\tau^*_i \ll 1$ [10]. From equation 5, it can be obtained that

$$\tau^*_i \cdot (1-p^*_i) \cdot T_{p,i} = \tau^*_j \cdot (1-p^*_j) \cdot T_{p,j} \quad (12)$$

Under the assumption that $\tau^*_i \ll 1$, from equation 4, we have

$$p^*_i \approx p^*_j \quad (13)$$

Therefore, we can make the following approximation,

$$\tau^*_i \cdot T_{p,i} \approx \tau^*_j \cdot T_{p,j} \quad (14)$$

After substituting $\Phi(\tau^*_{1,...,M}, p^*_{1,...,M})$ into equation 3, we have

$$\frac{1-P_{VTSS,i}}{1-P_{VTSS,j}} \approx \frac{\tau_i^* \cdot W_i}{\tau_j^* \cdot W_j} \tag{15}$$

Obviously, when the minimum contention window sizes W_i s are large, both $P_{VTSS,i}$ and p_i are small. Therefore, according to equation 11, the packet delay can be approximated as

$$T_{d,i} \approx \frac{n_i P_L}{S_i} \cdot (1-P_{VTSS,i}) \tag{16}$$

Based on equations 14, 15 and 16, it can be obtained that

$$\frac{T_{d,i}}{T_{d,j}} \approx \frac{W_i}{W_j} \tag{17}$$

Note that the above approximation only holds in the case that $P_{VTSS,i}$ s are very small. Equation 17 are exactly the same as the approximated results given in [9], which shows that saturation states can be regarded as an extreme case for non-saturation state with $P_{VTSS,i}=0$.

In the following part of this section, we try to find out how to properly set the minimum contention window sizes W_i s so as to achieve the target packet delay requirements, that is, $T_{d,i} < \hat{T}_{d,i}$. Combing equations 3 and 11, we have

$$T_{d,i} \approx \frac{n_i P_L}{S_i} + \left(\frac{P_s}{1-p_i} - \frac{n_i P_L}{S_i}\right)\left(1 - \frac{(1-2p_i)+p_i[1-(2p_i)^{m_i}]}{2(1-2p_i)} W_i \tau_i\right) \tag{18}$$

As we have already mentioned before, for stability, the system should operate near $\Phi(\tau^*_{1,...,M}, p^*_{1,...,M})$. Therefore, if it is required that $T_{d,i} < \hat{T}_{d,i}$, based on above equation

$$W_i < \left(\hat{T}_{d,i} - \frac{P_s}{1-p_i^*}\right) \Big/ \gamma^* \tag{19}$$

where $\gamma^* = \left(\frac{n_i P_L}{S_i} - \frac{P_s}{1-p_i^*}\right) \cdot \frac{(1-2p_i^*)+p_i^* \cdot [1-(2p_i^*)^{m_i}]}{2(1-2p_i^*)} \tau_i^*$. Equation 19 tells one the upper bounds for W_i s so as to meet the packet delay requirements $\hat{T}_{d,i}$ s.

4 Results and Discussions

In this section, both numerical and simulation results are shown to validate our proposed analysis model. In our experiments, the parameters for the system, which are based on IEEE 802.11b, are summarized as follows: MAC Header = 272 bits; PHY Header = 192 μs; ACK = 112bits + PHY Header; Channel Bit Rate = 11Mbps; Propagation Delay = 1 μs; Slot Time = 20 μs; SIFS = 10 μs; and DIFS = 50 μs. In our discrete-event simulation, a single-hop wireless LAN is considered. In the system, there are n_1 and n_2 type-1 and type-2 sending stations, respectively. Each of them carries only one traffic flow. It is assumed that the channel conditions are ideal (i.e., no hidden terminals and capture).

In the first experiments, two types of traffic flows are considered. Type-1 traffic has priority over type-2 traffic. Therefore, a smaller minimum contention window size W_1 is assigned to type-1 traffic, and a larger minimum contention window size $W_2 = 5W_1$ is allocated to type-2 traffic. Equations 3 and 5 are fundamental in this paper, they are validated in Fig. 2 and Fig. 3, respectively. In Fig. 2, the virtual time slot rates $P_{VTSS,i}$ s versus W_1 are shown. $P_{VTSS,i}$ s are obtained in two ways: one is by simulations. In the second way, packet collision rates p_i s and packet sending rates τ_i s, which are obtained by using simulations, are substituted into equation (3) to calculate the corresponding $P_{VTSS,i}$ s. System parameters are shown in the figure. It can be seen that the $P_{VTSS,i}$ s obtained by using equation 3 are very close to the simulated values, which validates the Markov model shown in Fig. 1. It can also be seen that when the minimum contention window sizes W_i s are very small, the differences between the simulated values and the estimated ones are larger. This is because in this case the packet collision rates increase dramatically and the behavior of the system is unstable.

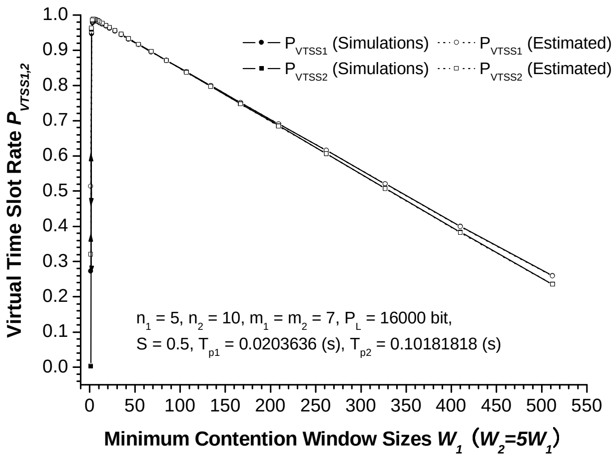

Fig. 2 Virtual time slot rates $P_{VTSS,i}$ versus W_1

In Fig. 3, the throughput S_i s are obtained by using two ways: one is by simulation. In the second way the throughputs are obtained by substituting p_i s and τ_i s, which are obtained from simulations, into equation 5. Again, it can be seen that when W_i s are very small, equation 5 can not describe the behavior of the system well, which is caused by the instability of the system behavior in this case.

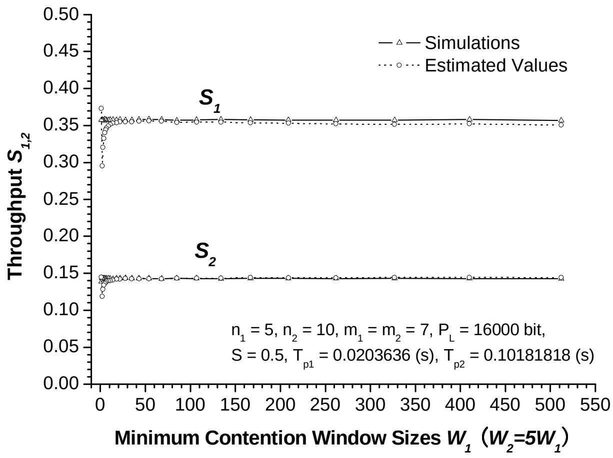

Fig. 3 Throughput S_i versus W_1

In Fig. 4, packet delays $T_{d,i}$ s versus W_i s are shown. Two ways are used to obtain $T_{d,i}$ s. One is through simulations. The other way is that we estimate $T_{d,i}$ s based on equation 11. In order to use equation 11, the solution $\Phi(\tau_1^*, \tau_2^*, p_1^*, p_2^*)$ is calculated by using equation 5 with the assumption that a_i and β are equal to 1. Then the obtained solution is substituted into equation 11. From the figure, it can be seen that when W_i s are not very small, packet delays can be successfully estimated. In this case, packet delays decrease linearly with the decrease of W_i s (we say that the system operates in "Stable State"). In this case, with the decrease of W_i s, time wasted in backoff processes can be directly converted into VTS without causing significant increase in packet collision rates and packet sending rates. When W_i s are very small, the behavior of the system is unstable (packet collision rates and packet sending rates increase drastically with the slight decrease of W_i s). Therefore, packet delays tend to increase drastically. In this case, the estimations of packet delays are not accurate. However, equation 11 is useful, because one does not want the system to operate far from the "Stable State". However, an interesting future research topic is to guarantee that the system operates under the "Stable State".

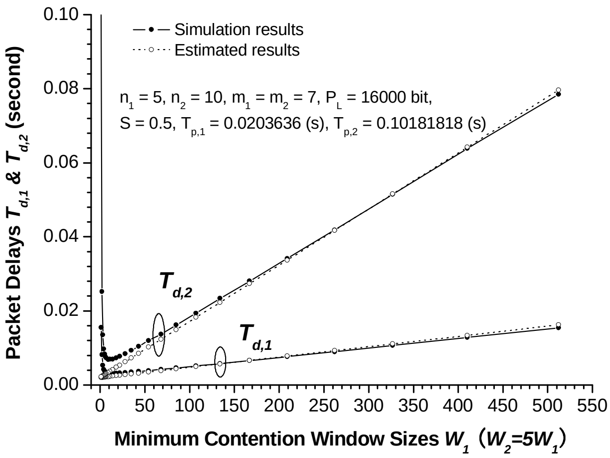

Fig. 4 Packet delays $T_{d,i}$ versus W_1

In Table 1, we demonstrate a possible application for our analysis model. In equation 19, we propose a way to estimate the upper bounds for the minimum contention window sizes W_i s to meet the required packet delays $T_{d,i} < \hat{T}_{d,i}$. In this example, we first estimate the upper bounds for W_i s based on equation 13. Then, to better understand the performance of the estimated upper bounds, actual packet delays $T_{d,i}$ s are obtained from simulations with the corresponding W_i s being set to be equal to the corresponding estimated upper bounds. Finally, comparisons can be easily made by comparing the obtained packet delays $T_{d,i}$ s and the required packet delays $\hat{T}_{d,i}$ s. In Table 1, the first two columns are the packet delay requirements. The third and fourth columns are estimated minimum contention window sizes by using equation 19. The last two columns are the achieved packet delays obtained from simulations. It can be seen that the packet delay requirements can be approximately met, which suggests a promising application for our proposed model.

Table I. Guarantee Packet Delay Requirements

$\hat{T}_{d,1}$	$\hat{T}_{d,2}$	W_1	W_2	$T_{d,1}$	$T_{d,2}$
0.005	0.030	109	924	0.005052	0.030827
0.005	0.025	109	759	0.005085	0.025859
0.005	0.020	109	594	0.005126	0.020966
0.005	0.015	109	429	0.005204	0.016293
0.005	0.010	109	264	0.005313	0.011022
0.005	0.005	109	99	0.005504	0.005506

System parameters: $P_{Len,1}$ = $P_{Len,2}$ = 2000 bytes, n_1 = 5, n_2 = 10, m_1=m_2=7, $T_{P,1}$ = 0.020363636 s, $T_{P,2}$ = 0.10181818 s

5 Conclusions

In this paper, a simple model has been proposed to analyze the performance of IEEE 802.11 DCF with service differentiation support in non-saturation states, which helps one to obtain deeper insight into the IEEE 802.11 DCF. Under the case that the system operates in stable states, we can approximately evaluate the most important system performance measures, such as packet delays, which provide one with an important tool to predict the system performance. Moreover, in order to meet certain packet delay requirements, a practical method has been given based on our theoretical results. Comparisons with simulation results show that this method does achieve the specified packet delay requirements with good accuracy. Possible extensions of this work to consider practical schemes capable to rapidly adapt to changing traffic loads are now being considered.

References

1. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, IEEE Standard 802.11, Aug. 1999.
2. T. S. Ho and K. C. Chen, “Performance evaluation and enhancement of the CSMA/CA MAC protocol for 802.11 wireless LAN’s,” Proceedings of IEEE PIMRC, Taipei, Taiwan, Oct. 1996, pp.392-396.
3. F. Cali, M. Conti, and E. Gregori, “IEEE 802.11 wireless LAN: Capacity analysis and protocol enhancement,” Proceedings of INFOCOM’98, San Francisco, CA, March 1998, vol. 1, pp. 142 -149.
4. G. Bianchi, “Performance analysis of the IEEE 802.11 distributed coordination function,” IEEE Journal on Selected Areas In Communications, vol. 18, no. 3, March 2000.
5. E. Ziouva and T. Antonakopoulos, “CSMA/CA performance under high traffic conditions: Throughput and delay analysis,” Computer Communications, vol. 25, pp. 313–321, 2002.
6. Foh, C.H., Zukerman, M., “Performance Analysis of the IEEE 802.11 MAC Protocol,” Proc. European Wireless, 2002.
7. E.M.M. Winands, T.J.J. Denteneer, J.A.C. Resing and R. Rietman, “A finite-source feedback queueing network as a model for the IEEE 802.11 Distributed Coordination Function,” Proc. European Wireless, 2004.
8. O. Tickoo and B. Sikdar, “Queueing Analysis and Delay Mitigation in IEEE 802.11 Random Access MAC based Wireless Networks,” in Proceedings of IEEE INFOCOM'04, Hong Kong, China, March 2004.
9. LI Bo, Roberto Battiti, “Performance Analysis of An Enhanced IEEE 802.11 Distributed Coordination Function Supporting Service Differentiation,” QoFIS 2003, Sweden, Springer LNCS volume 2811, pp. 152-161.
10. LI Bo and Roberto Battiti, “Achieving Maximum Throughput and Service Differentiation by Enhancing the IEEE 802.11 MAC Protocol,” WONS 2004 (Wireless On-demand Network Systems, Trento, Italy), Springer Lecture Notes on Computer Science LNCS volume 2928, pp. 285-301.
11. W. G. Marchal, “An approximate formula for waiting time in single server queues,” AIIE Transactions, Dec. 1976, 473-474.