

Dynamic Transmission Control Method for WLANs*

JAIN-SHING LIU AND CHUN-HUNG RICHARD LIN[†]
*Department of Computer Science and Information Engineering
Providence University
Taichung, 433 Taiwan*

[‡]*Department of Computer Science and Engineering
National Sun Yat-Sen University
Kaohsiung, 804 Taiwan*

Due to the popularity of wireless local area networks (WLANs), many research works for such networks have been done, mainly based on the assumption that the number of nodes, N , is known a priori and remains the same in the runtime. However, in the real world, N changes dynamically because wireless stations can be turned on/off or moved in/out anytime and anywhere. As a consequence, the theoretical results based on the assumption usually have their own limitations when considered with industrial applications in a WLAN. For this problem, we introduce in this paper a dynamic transmission control method to take into account the dynamic of WLANs. Based on the so-called p-persistent IEEE 802.11, the dynamic method can adaptively adjust the transmission probability in each back-off stage of DCF to achieve the maximum performance in the runtime. We use analysis to show how the method can achieve the maximal performance in theory. With simulations, we further confirm these results and show how the parameters involved can be settled to achieve the performance aims.

Keywords: dynamic transmission control, WLANs, IEEE 802.11 DCF, throughput maximization, performance analysis

1. INTRODUCTION

IEEE 802.11 [1] has become one of the most important and successful MAC protocols for wireless infrastructure and infrastructureless (*i.e.*, ad hoc) LANs. In this standard, Distributed Coordination Function (DCF) is mainly used as primary mechanism to access the medium. This function adopts a carrier sense multiple access with collision avoidance (CSMA/CA) with the binary exponential back-off (BEB) procedure, which enables fast installation with minimal management and maintenance costs in WLANs, just as the most successful wired counterpart, Ethernet.

For the DCF, many research works have been done, mainly based on the assumption that the number of nodes, N , is known a priori and remains the same in the runtime. However, several studies [2-5] have shown that the performance evaluation of DCF is very sensitive to the number of nodes competing on the medium access. And in the real world, this number will vary dynamically for the wireless stations involved can be turned on/off or moved in/out anytime and anywhere. Consequently, the theoretical results based on the assumption usually have their own limitations when considered with real applications in a WLAN.

Received December 15, 2006; revised April 26, 2007; accepted June 26, 2007.

Communicated by Yu-Chee Tseng.

* This work was supported by the National Science Council of Taiwan, R.O.C., under grant No. NSC 95-2221-E-126-002.

For this problem, we introduce here a dynamic transmission control method to take into account the dynamic of WLANs. Based on the so-called p-persistent IEEE 802.11, the method can adaptively adjust the transmission probability in each back-off stage of DCF to achieve the maximum performances in the runtime. In addition, to get rid of the cumbersome problem of obtaining the number of nodes in a WLAN that is usually required by an optimal method in theory to actually work, we use the so-called Slot Busy Utilization (SBU) to indicate the network congestion level. With this metric, each node can independently choose the optimal p-persistent transmission probability to filter its data transmissions in spite of the actual number of competing nodes. In particular, the optimal probability can be calculated in advance and stored in a table, which shifts the computation burden offline, and hence, significantly simplifies the runtime execution.

The paper is organized as follows. In section 2, we first present the background and related works of this paper. Then, in section 3, we briefly introduce and review the p-persistent transmission control protocol and the corresponding Markov model analysis. According to this model, the optimal transmission probability for throughput is derived in section 4, which then provides in section 5 the basis of constructing the dynamic transmission control method. Finally, the analytical results are examined with experiments in section 6, and conclusions are drawn in section 7.

2. BACKGROUND AND RELATED WORKS

For WLANs, there are some related works being done on the performance issues of IEEE 802.11 DCF [6-10]. For example, in [6], a multiplicative increase, linear decrease (MILD) algorithm is introduced to prevent the oscillation of the contention window in BEB. This algorithm increases the contention window by 1.5 times when collision occurs, and decreases the window by 1 when transmission succeeds. In [7], an exponential increase, exponential decrease (EIED) algorithm is proposed to increase the contention window by a multiple when collision occurs and to exponentially decrease the window when transmission succeeds. In [8], a linear increase, linear decrease (LILD) algorithm is given to always adjust the window size by a constant value. Besides, a so-called GDCF algorithm is given in [9], which doubles the window size after each unsuccessful transmission and halves this size after c consecutive successful transmissions. Recently, in [10], the authors propose a multi-chain backoff (MCB) algorithm that enables stations to adapt to different congestion levels by exploiting multiple back-off chain.

Unlike the previous works that adjust the parameters of DCF, *e.g.*, the contention window size, in this work we use a p-persistent enhanced IEEE 802.11 protocol, namely *P-IEEE 802.11 DCF*, that directly manipulates the transmission probability of a node to improve the performance of IEEE 802.11 WLANs. Based on the protocol, we consider a dynamic transmission control method that can self-adapt the contention level by using the so-called SBU derived from optimal performance calculations for WLANs. In particular, the method can achieve the aim without relying on the actual number of nodes competing the medium access. In addition, by calculating the optimal probability in advance and storing in a table, we can shift the computation burden offline and significantly simplifies the runtime execution. These characteristics make the introduced method more realistic than certain previous works when considered with real applications in WLANs.

3. P-IEEE 802.11 DCF

In this section, we introduce the so-called P-IEEE 802.11 DCF and its Markov chain model. With this model, we can obtain the optimal transmission probability shown in the next section.

3.1 Overview of P-IEEE 802.11 DCF

The operations of P-IEEE 802.11 DCF are summarized in Fig. 1. As can be seen, the core idea of this DCF is simply inserting a separate layer between the standard access scheme and the physical layer. This layer calculates the opportunity to perform a transmission attempt. It is used by a node to decide whether to perform or to defer the scheduled transmission attempt. When carried out, each node filters its transmission attempt based on the decision made in this layer, *i.e.*, the p-persistent transmission probability, P^t . That is, each node draws a random number X , and if $X \leq P^t$, the frame under consideration is transmitted by the legacy DCF. Otherwise, if $X > P^t$, the frame is deferred with a new back-off interval, just as that it encounters a collision in the MAC.

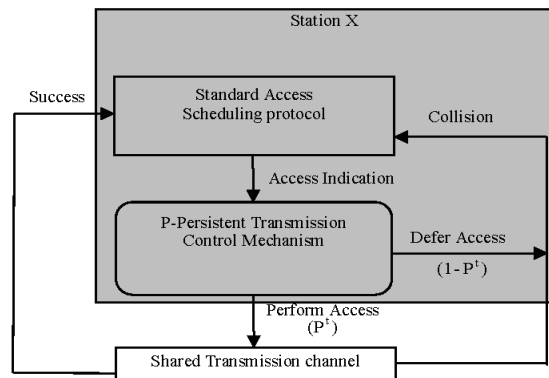


Fig. 1. The architecture of P-IEEE 802.11 DCF.

As described in above, this protocol follows the conventional p-persistent CSMA. In addition, with this protocol each node can filter its transmissions based on some meaningful criteria (*e.g.*, SBU to be introduced) that can properly manipulate P^t to achieve the maximum system performance.

3.2 Throughput Calculation

Given the above overview, it still remains a major factor to be determined: how a node can decide the controlled p-persistent transmission probability to achieve certain performance requirements. To this end, this probability, called P_T for reference, should be represented with a form that can be analyzed easily. In this work, this is done with a simple non-uniform increasing function that can reasonably reflect the channel contention level sensed by a node in back-off stage i . That is, the probability is given by

$$P_i^t = 1 - \phi^{i+1} \tag{1}$$

where ϕ denotes the so-called transmission factor.

Given P_i^t in Eq. (1) and other parameters: m (the maximum back-off stage), N (the number of nodes in the network) and W_i (the back-off window size for stage i), we can calculate the saturated throughput of P-IEEE by using a Markov chain analysis that takes P_Ts into account, based on the model extended from [4]. As a brief summary, the non-null probabilities involved in the Markov chain, as shown in Fig. 2, can be given by

$$\left\{ \begin{array}{l} P\{i, k | i, k+1\} = 1, \quad k \in [0, W_i - 2], i \in [0, m] \\ P\{0, k | i, 0\} = \frac{(1-P) \cdot P_i^t}{W_0}, \quad k \in [0, W_0 - 1], i \in [0, m] \\ P\{i, k | i-1, 0\} = \frac{1 - (1-P) \cdot P_{i-1}^t}{W_i}, \quad k \in [0, W_i - 1], i \in [1, m] \\ P\{0, k | m, 0\} = \frac{1}{W_0}, \quad k \in [0, W_m - 1] \end{array} \right. \tag{2}$$

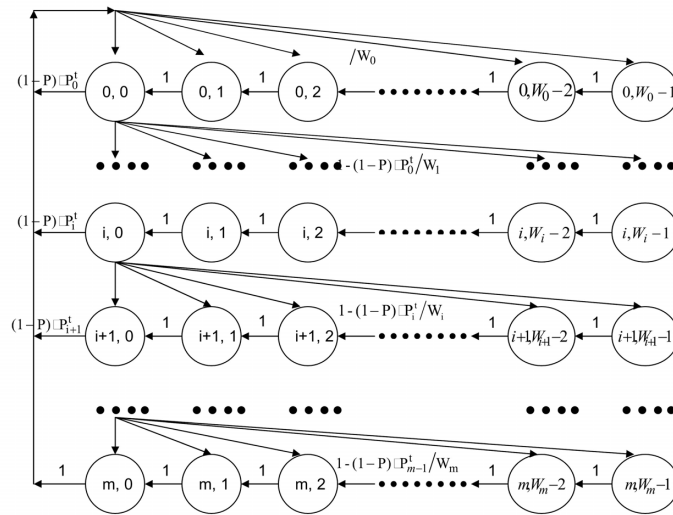


Fig. 2. The Markov chain model for the P-IEEE 802.11 MAC.

With some manipulations, we can lead the above to the stationary probability, $b_{i,k}$, for a node with back-off stage i and back-off timer k ,

$$b_{i,k} = \frac{W_i - k}{W_i} \cdot \begin{cases} (1-P) \cdot \sum_{i=0}^{m-1} P_i^t \cdot \prod_{j=0}^{i-1} (1 - (1-P)P_j^t) \cdot b_{0,0} + b_{m,0}, & i = 0 \\ (1 - (1-P)P_{i-1}^t) \cdot b_{i-1,0}, & 0 < i \leq m \end{cases} \tag{3}$$

Finally, the probability τ that a node transmits a frame in a randomly chosen time and the probability P that a station in the back-off stage senses the channel busy constitute a non-linear system of equations as follow

$$\begin{cases} \tau = \sum_{i=0}^m P_i^t \cdot b_{i,0} \\ P = 1 - (1 - \tau)^{N-1} \end{cases} \quad (4)$$

Further, for the throughput calculation, we let P_{tr} be the probability of at least one transmission in a slot time, and P_s the probability that a successful transmission occurs in a slot time. In terms of τ , these probabilities can be obtained as $P_{tr} = 1 - (1 - \tau)^N$ and $P_s = N \cdot \tau (1 - \tau)^{N-1} / P_{tr}$. With these probabilities, we can express the overall system throughput, S , as the following ratio,

$$S = \frac{P_s \cdot P_{tr} \cdot E[P]}{(1 - P_{tr}) \cdot \sigma + P_{tr} \cdot P_s \cdot T_s + P_{tr} \cdot (1 - P_s) \cdot T_c} \quad (5)$$

where $E[P]$ denotes the average frame length, σ the duration of an empty slot time, and T_s and T_c the average times the channel is sensed busy due to a successful transmission or a collision, respectively. For the values of these parameters, one may refer to [4].

4. OPTIMAL P-PERSISTENT TRANSMISSION PROBABILITY

According to the throughput analysis in the previous section, we now derive the optimal transmission probability for each back-off stage P_i^* in terms of the transmission factor ϕ^* . For doing so, we first replace the probabilities in the throughput calculation of Eq. (5) with more concise notations. That is, we use the following probabilities: $P_{coll} = P_{tr} \cdot (1 - P_s) = 1 - N \cdot \tau \cdot (1 - \tau)^{N-1} - (1 - \tau)^N$, $P_{free} = 1 - P_{tr} = (1 - \tau)^N$, and $P_{succ} = P_{tr} \cdot P_s = N \cdot \tau \cdot (1 - \tau)^{N-1}$ to represent the same Eq. (5). Then, we multiply the resulting expression by $1/(P_{succ} \cdot \sigma)$ that removes the probability P_{succ} in the numerator part and replaces all the periods of time involved with those measured in slots (T_s^σ and T_c^σ), leading to

$$\begin{aligned} S &= \frac{P_{succ} \cdot E[P]}{P_{free} \cdot \sigma + P_{succ} \cdot T_s + P_{coll} \cdot T_c} \times \frac{1}{P_{succ} \cdot \sigma} = \frac{\frac{E[P]}{\sigma}}{\frac{T_s}{\sigma} + \frac{P_{free}}{P_{succ}} + \frac{P_{coll}}{P_{succ}} \cdot \frac{T_c}{\sigma}} \\ &= \frac{\frac{E[P]}{\sigma}}{T_s^\sigma + \frac{(1 - \tau)^N}{N \cdot \tau \cdot (1 - \tau)^{N-1}} + \frac{1 - (1 - \tau)^{N-1} - N \cdot \tau \cdot (1 - \tau)^{N-1}}{N \cdot \tau \cdot (1 - \tau)^{N-1}} \cdot T_c^\sigma}. \end{aligned} \quad (6)$$

From the last equation, we can see that if the denominator part is minimized, the throughput S is maximized. Thus, we define this part except for the constant time, T_s^σ , as

$$\begin{aligned}
S_f(\tau) &= \frac{(1-\tau)^N}{N \cdot \tau \cdot (1-\tau)^{N-1}} + \frac{1 - (1-\tau)^{N-1} - N \cdot \tau \cdot (1-\tau)^{N-1}}{N \cdot \tau \cdot (1-\tau)^{N-1}} \cdot T_c^\sigma \\
&= \frac{(1-\tau)^N + [1 - (1-\tau)^{N-1} - N \cdot \tau \cdot (1-\tau)^{N-1}] \cdot T_c^\sigma}{N \cdot \tau \cdot (1-\tau)^{N-1}}.
\end{aligned} \tag{7}$$

Taking the derivation of function $S_f(\tau)$ with respect to τ , and letting $S'_f(\tau) = 0$, we get

$$(1-\tau)^N + T_c^\sigma \cdot [1 - N \cdot \tau - (1-\tau)^N] = 0. \tag{8}$$

Under the condition, $\tau \ll 1$, $(1-\tau)^N \approx 1 - N \cdot \tau + \frac{N \cdot (N-1)}{2} \cdot \tau^2$, the solution of Eq. (8) can be approximated by

$$\tau_{opt} = \frac{\sqrt{\frac{N + 2 \cdot (N-1)(T_c^\sigma - 1)}{N}} - 1}{(N-1)(T_c^\sigma - 1)}. \tag{9}$$

The result of this approximation is then used to be the initial value in the non-linear optimization for Eq. (8), which provides a more accurate solution of τ_{opt} . Consequently, the value of τ_{opt} is taken into Eqs. (4) and (5) for getting the optimal transmission factor ϕ^* that can achieve the maximum throughput. That is,

$$\phi^* = \arg \text{MAX}_{\{\phi^*\}} \{S(\tau_{opt})\}. \tag{10}$$

Taking this factor into Eq. (1), we can finally obtain the optimal transmission probability, P_i^{I*} , as required.

5. DYNAMIC TRANSMISSION CONTROL METHOD

In this section, we turn our attention to the dynamic transmission control method, based on the P-IEEE 802.11 DCF. By adaptively adjusting the transmission probability with respect to the measured channel contention level, the method can achieve the maximum performances in the runtime, and requires no exact knowledge about the number of competing nodes in a WLAN.

5.1 Quasi-Invariant in IEEE 802.11 WLAN

To show how the method can be done, we first introduce a quasi-invariant in the IEEE 802.11 WLAN. The quasi-invariant is inspired by the fact that if N in a WLAN is large, $N-1$ in Eq. (9) can be approximated by N . The results in an approximated optimal τ as

$$\tau_{opt} \approx \frac{\sqrt{\frac{N + 2N \cdot (T_c^\sigma - 1)}{N}} - 1}{N \cdot (T_c^\sigma - 1)}. \tag{11}$$

When multiplied by N and then applied to further approximation, it becomes

$$N \cdot \tau_{opt} = \frac{\sqrt{\frac{N+2N \cdot (T_c^\sigma - 1)}{N}} - 1}{T_c^\sigma - 1} \approx \frac{-1 + \sqrt{\frac{N+2N \cdot T_c^\sigma}{N}}}{T_c^\sigma} \approx \frac{-1 + \sqrt{1+2 \cdot T_c^\sigma}}{T_c^\sigma}. \quad (12)$$

As shown in the last equation, the product of N and τ_{opt} can be approximated by a value independent of both, representing a quasi-invariant in the WLAN.

5.2 Slot_Busy_Utilization

With the quasi-invariant in above, we now try to estimate the network congestion level. For doing so with low costs, we consider a simple method that involves zero costs and has no overheads on the protocol performance. To fulfill the requirement, the information involved should be already available in the IEEE 802.11 Standard. To this end, we define the Slot_Busy_Utilization, SBU, as

$$SBU = \frac{\text{Busy_Time}}{\text{Available_Time}}. \quad (13)$$

In above, the available period of time (Available_Time) is accounted as the whole period of time a back-off takes to resolve a collision, and the busy period of time (Busy_Time) is accounted as the sum of the periods of time encountered by a collision or a successful transmission during the back-off. Both are available with the legacy MAC, and thus satisfy our requirements.

In addition to its simplicity, SBU is considered here due to its analytical representation. Specifically, SBU can be represented in terms of the periods of time involved, T_s , T_c and σ , proportional to their corresponding probabilities, P_{succ} , P_{coll} , and P_{free} , respectively, by

$$\begin{aligned} SBU &= \frac{P_{succ} \cdot T_s + P_{coll} \cdot T_c}{P_{succ} \cdot T_s + P_{free} \cdot \sigma + P_{coll} \cdot T_c} \\ &= \frac{T_s^\sigma + \frac{1 - (1-\tau)^N - N \cdot \tau \cdot (1-\tau)^{N-1}}{N \cdot \tau \cdot (1-\tau)^{N-1}} \cdot T_c^\sigma}{T_s^\sigma + \frac{(1-\tau)^N}{N \cdot \tau \cdot (1-\tau)^{N-1}} + \frac{1 - (1-\tau)^N - N \cdot \tau \cdot (1-\tau)^{N-1}}{N \cdot \tau \cdot (1-\tau)^{N-1}} \cdot T_c^\sigma}. \end{aligned} \quad (14)$$

The last Eq. (14) is obtained with some simple substitutions, and involves only the periods of time mentioned previously, the number of nodes, N , and the transmission probability, τ . With this equation, we can easily see that the optimal SBU (SBU^*) for the maximum system performance depends on the number of nodes in the network, N , and the optimal transmission probability, τ_{opt} . Thus, N and τ_{opt} should be found at first, for the calculation of $SBU^* = SBU(N, \tau_{opt})$. However, it is known that N usually requires sophisticated mechanisms to estimate its value robustly [11], and τ_{opt} also depends on N . To alleviate the cumbersome problem, in our work, instead of directly measuring N and then

calculating τ_{opt} , we take another metric to measure the network congestion level. This is done by minor modifications to Eq. (14), and by the fact that $N \cdot \tau_{opt}$ is a quasi-invariant to replace the corresponding terms in Eq. (14). Thus, we have the following approximation of SBU^* ,

$$SBU_{app}^* = \frac{T_s^\sigma + \frac{1 - (1 - N \cdot \tau_{opt}) - N \cdot \tau_{opt} \cdot (1 - N \cdot \tau_{opt})}{N \cdot \tau_{opt} \cdot (1 - N \cdot \tau_{opt})} \cdot T_c^\sigma}{T_s^\sigma + \frac{(1 - N \cdot \tau_{opt})}{N \cdot \tau_{opt} \cdot (1 - N \cdot \tau_{opt})} + \frac{1 - (1 - N \cdot \tau_{opt}) - N \cdot \tau_{opt} \cdot (1 - N \cdot \tau_{opt})}{N \cdot \tau_{opt} \cdot (1 - N \cdot \tau_{opt})} \cdot T_c^\sigma} \quad (15)$$

which involves only the three constant periods of time and the quasi-invariant, and all of them can be obtained a priori. This significantly reduces the computation complexity for SBU^* .

```

SBUn = current SBU measurement
tryn = current # of retry in the IEEE 802.11 MAC
WIN = measurement window size
MAX = measurement threshold
ρ = offset factor

Begin
  // Step 1: update the optimal transmission factor
  count = count + 1;
  if SBUn ≥ (1 + ρ) · Optimal_SBU_Table(rate)
    α = α + 1;
  else if SBUn ≤ (1 + ρ) · Optimal_SBU_Table(rate)
    α = α - 1;
  end
  if count ≥ WIN
    if α ≥ MAX
      # node = # node of the next higher scale;
    else if α ≤ -MAX
      # node = # node of the next lower scale;
    end
    reset count and α;
  end
  ϕ* = Optimal_Transmission_Table(# node, rate);
  // Step 2: transmit frames with the P-persistent rule
  obtain a random value P, 0 ≤ P ≤ 1
  if P < P'tryn = 1 - ϕ*tryn+1 or tryn > 802_11_Retry_Limit
    send this frame;
  else
    tryn+1 = tryn + 1;
    increase the contention window;
    go to the backoff stage;
  end
end

```

Fig. 3. Pseudo code segment for the dynamic step method.

5.3 Dynamic Step Method

With the aid of the quasi-invariant and SBU derived above, the so-called dynamic step method is introduced to accommodate the dynamic of WLANs. As shown in Fig. 3, the method is table-driven. That is, before running this method, we offline calculate and collect ϕ^* s into the Optimal_Transmission_Table for each scale of N and each data rate available in the IEEE 802.11 PHY, and collect SBU_{app}^* s into the Optimal_SBU_Table for each data rate. In above, instead of using every number of nodes, we consider a quantized number with step size of 10 (*i.e.*, 10, 20, ..., *etc.*). It is adopted here because ϕ^* s for two neighboring numbers of nodes usually have very close values, while those from the quantized scale can almost provide distinguishable differences. Tuning ϕ^* according to the latter has the merits of significantly saving computation costs, resulting in a more stable ϕ^* s in the runtime, and remaining similar performance benefits, when compared with the former.

Given the above, the runtime behavior of a node is to measure its own SBU and compares this value with SBU_{app}^* stored in its table. If the measurement has a value lower than the latter, the network is considered under-saturated; otherwise, it is considered stable or over-saturated. More precisely, each node maintains two variables, *count* and α , to dynamically reflect the network congestion level. In addition, let *WIN* be the number of transmissions to be observed by a node in a cycle, *MAX* be the threshold to certify a change of the network congestion level, and ρ be a pre-designed parameter to smoothly adapt to the congestion level. As shown in Fig. 3, when a node transmits a data frame, *count* will increase by 1 to accumulate the number of transmission already performed for a node in a cycle. At the same time, if SBU can not correctly reflect the network congestion level (the number of competing nodes in the network), the value of SBU will become larger or smaller than the corresponding optimal value, SBU^* , by a factor of $(1 + \rho)$, *i.e.*, $SBU > (1 + \rho) \cdot SBU_{app}^*$ or $SBU < (1 - \rho) \cdot SBU_{app}^*$, with a pre-defined ρ . To smoothly reflect the network dynamic, instead of immediately adjusting the transmission factor, ϕ , a node records the number of this event into α by increasing or decreasing the variable by 1. When the value of α is higher (lower) than the predefined threshold +MAX (−MAX) in a cycle of *WIN* observations, a larger (smaller) value of ϕ^* will be retrieved from the pre-calculated Optimal_Transmission_Table to reflect the increased (decreased) N realized by a node. After that, a node resets *count* and α to 0 when a cycle of *WIN* observations is over, and then, starts a new observation window.

As can be seen above, the dynamic step method is intended to smoothly adapt to the network congestion level for achieving the maximum system performance in a WLAN. One important aspect of this algorithm is that it shifts the computation burden offline, and hence, significantly simplifies the runtime execution. Therefore, embedding the algorithm at the MAC layer has little effect on the performance of higher-layer applications, which is a desirable feature for any MAC-layer enhancement. More importantly, this algorithm can efficiently accommodate the dynamic of WLANs, and provide nearly the same performance of the optimal method in theory.

6. PERFORMANCE EVALUATION

In this section, we report on experiments made in order to verify the performance

results considered previously. To this end, we simulate a WLAN with N ranging from 10 to 100 (with step size of 10), which reasonably represents the possible population resident in a WLAN. The IEEE 802.11 MAC/PHY is adopted in the experiments, wherein all control frames are transmitted using the lowest rate in the basic rate set for the IEEE 802.11a PHY to ensure that the most distant node in the WLAN can reach AP. In addition, all nodes are uniformly distributed in the WLAN, and each node has a continuous CBR flow with 1,024-bytes of UDP packets toward AP, which properly saturates the network. With the assumption of no hidden terminals, the basic mode is considered with all 8 different data rates in IEEE 802.11a PHY. However, we show only the result for the lowest data rate of 6Mbps because of space limitations. The results for the other rates have the same trend.

In addition, we aim to verify the performance results with different parameters involved in the dynamic step method. Thus, we set $WIN = 10$ with $MAX = 8$, and $WIN = 100$ with $MAX = 80$ to account for a shorter cycle and a longer one, respectively, while remaining the same ratio between the two parameters. In addition, ρ is given with value of 0.1, 0.05, and 0.01, respectively, to account for the possible difference between the SBU measured by a node and the optimal SBU^*_{app} in the theory. Each setting is carried out

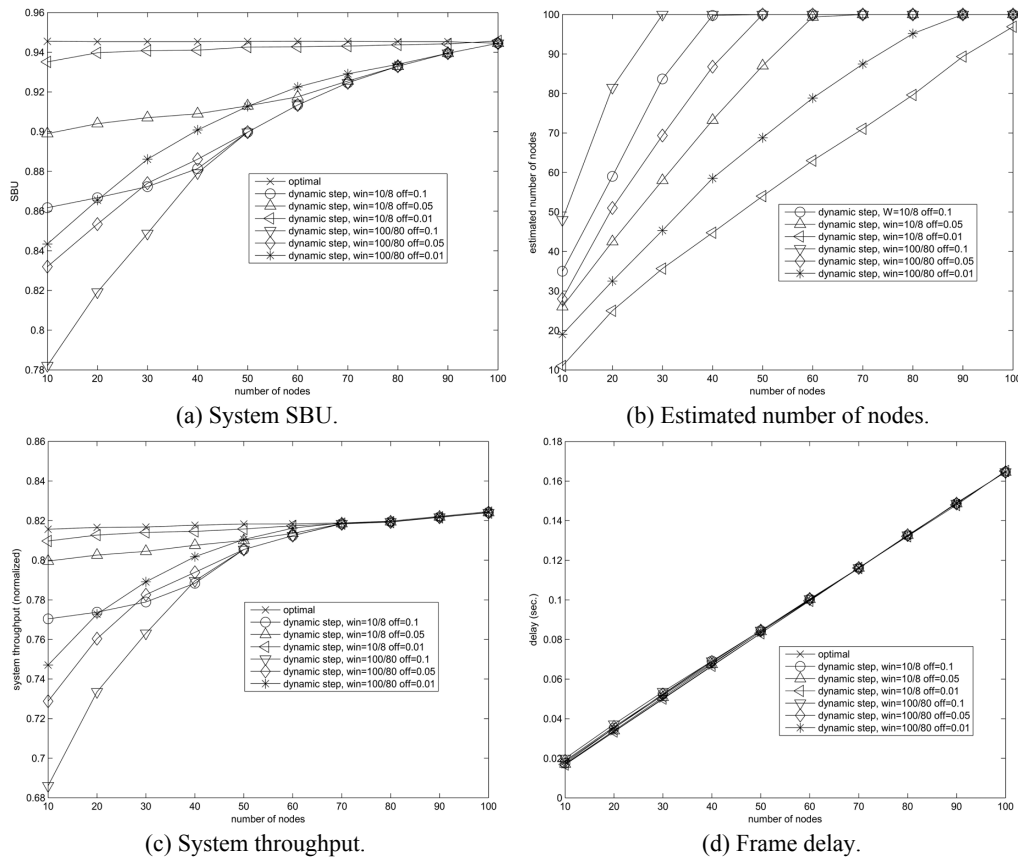


Fig. 4. Performance results for different parameters.

20 times to obtain its mean value as the result. The results are shown in Fig. 4. As can be seen, the method with the shorter cycle, *i.e.*, $\text{win} = 10/8$, and the smallest $\rho = 0.01$, results in the best performance that is comparable to the optimal. This implies that with this method, a shorter cycle is enough for capturing the dynamic of WLANs in time, and SBU_{app}^* is an efficient approximation, leaving only little difference between its value and the measured one. This is evident in Fig. 4 (b), showing that the method with this setting yields the most correct estimation on N .

7. CONCLUSION

In this paper, we proposed and extensively evaluated a dynamic transmission control method based on the so-called *P-IEEE 802.11 DCF*. Specifically, we have shown the effectiveness of the dynamic method in terms of slot busy utilization (SBU), estimated number of nodes, system/per node throughput, and frame delay. The experiment results indicate that the method can simultaneously improve all the performance metrics and keep their values very near to the optimal. In addition, these results also confirm the robustness and insensitivity of the method by showing their stable performances in spite of N actually involved. This is particularly useful because unlike many previous works for WLANs, our method does not assume N being known a priori and remaining the same in the runtime. Thus, the proposed method has its own values not limited to that of theory, and can be actually considered with real industrial applications in a WLAN.

REFERENCES

1. IEEE 802.11 II – Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, Standard, IEEE, 1999.
2. J. Weinmiller, M. Schlager, A. Festag, and A. Wolisz, “Performance study of access control in wireless LANs IEEE 802.11 DCF and ETSI HiperLAN,” *Mobile Networks and Applications*, Vol. 2, 1997, pp. 55-67.
3. B. P. Crow, I. Widjaja, J. G. Kim, and P. T. Sakai. “IEEE 802.11 wireless local area networks,” *IEEE Communication Magazine*, 1997, pp. 116-126.
4. G. Bianchi, “Performance analysis of the IEEE 802.11 distributed coordination function,” *IEEE Journal on Selected Areas in Communications*, Vol. 18, 2003, pp. 535-547.
5. Y. C. Tay and K. C. Chua, “A capacity analysis for the IEEE 802.11 MAC protocol,” *ACM/Baltzer Wireless Networks*, Vol. 7, 2001, pp. 159-171.
6. V. Bharghvan, “MACAW: a media access protocol for wireless LANs,” in *Proceedings of ACM SIGCOMM*, 1994, pp. 212-225.
7. N. O. Song, B. J. Kwak, J. Song, and L. E. Miller, “Enhancement of IEEE 802.11 distributed coordination function with exponential increase exponential decrease backoff algorithm,” in *Proceedings of IEEE Vehicular Technology Conference*, Vol. 4, 2003, pp. 2775-2778.
8. V. Vistas, “Throughput analysis of linear backoff scheme in wireless LANs,” *IEEE Electronic Letters*, Vol. 39, 2003, pp. 99-100.
9. C. Wang, B. Li, and L. Li, “A new collision resolution mechanism to enhance the

performance of IEEE 802.11 DCF,” *IEEE Transactions on Vehicular Technology*, Vol. 53, 2004, pp. 1235-1246.

10. S. R. Ye and Y. C. Tseng, “A multi-chain backoff mechanism for IEEE 802.11 WLANs,” *IEEE Transactions on Vehicular Technology*, to appear.
11. G. Bianchi and I. Tinnirello, “Kalman filter estimation of the number of competing terminals in an IEEE 802.11 network,” in *Proceedings of INFOCOM*, Vol. 2, 2003, pp. 844-852.



Jain-Shing Liu (劉建興) was born in Taipei, Taiwan, in 1970. He received the Ph.D. degree in the Department of Computer and Information Science, National Chiao Tung University, Hsinchu, Taiwan. He is currently with the faculty of the Department of Computer Science and Information Engineering, Providence University, Taichung Taiwan. His research interests include wireless communication protocol design, personal communication network, and performance analysis of wireless networks. Dr. Liu is a member of IEEE and IEICE.



Chun-Hung Richard Lin (林俊宏) was born in Kaohsiung, Taiwan. He received the B.S. and M.S. degrees from the Department of Computer Science and Information Engineering, National Taiwan University, in 1987 and 1989, respectively, and the Ph.D. degree from Computer Science Department, University of California, Los Angeles (UCLA), in 1996.

Dr. Lin joined National Chung Cheng University in Taiwan in 1996. Since August 2000, he has been with the Department of Computer Science and Engineering, National Sun Yat-Sen University, Kaohsiung, Taiwan. His research interests include the design and control of personal communication networks, protocol design and implementation for differentiated/integrated services in mobile wireless networks, mobile Internet, distributed simulation, and embedded operating system design and implementation.