Optimal ATIM size for 802.11 networks in ad hoc mode

Xia Gao, Fujio Watanabe
DoCoMo Communications Lab USA

***Abstract*— Power management in an ad hoc WLAN (IBSS) is not well studied compared with infrastructure based WLAN and many open issues still exist. This paper provides methods to calculate the optimal ATIM window size in an IBSS WLAN. The derived equations capture the detail of power saving techniques used in current 802.11 standards. The length of different time periods including beacon transmission, ATIM transmission, and data transmission are all derived. This algorithm provides the basis to setup original ATIM window size based on the number of nodes in the system. With other dynamic adaptation schemes, it can achieve good performance while minimizing the power consumption. Our initial simulation results further prove the accuracy of the scheme.**

I. INTRODUCTION

Compared with power management in an infrastructure network, power management in an ad hoc WLAN (called IBSS in the standard) is not well studied and not very efficient. The main reason is that AP in an infrastructure WLAN has the global knowledge of power saving states of all mobile nodes associated with it. All communications to and from mobile nodes have to go through AP first. So AP is able to buffer packets for mobile stations in the power saving (PS) mode and notify these nodes during pre-specified time interval to receive packets. On the other hand, there is no entity in IBSS similar to AP that has global knowledge of power saving states of all nodes. As a result, each node has to store packets locally and communicate individually with its peers to schedule when to deliver packets.

One specific problem of power saving in IBSS is the optimal ATIM window size design. Current 802.11 standard requires the ATIM windows size to be a fixed size throughout the lifespan of an IBSS where the ATIM window size is determined by the STA initiating the IBSS. However, if the ATIM window size is too small, all the ATIM messages cannot be transmitted during the ATIM window. As a result, data frames of the un-announced traffic that could have been transmitted in the current beacon interval has to wait until the next beacon interval. This causes unnecessary delay and wastes channel bandwidth. On the other hand, the ATIM window size cannot be too long. As the ATIM window size increases, there is a corresponding decrease in the time left in the current Beacon Interval for data transmissions. If the ATIM window size becomes too large, a good portion of the time towards the end of ATIM window is idle or power-on nodes may not have enough time to send packets during the left-over time. This incurs both power waste and bandwidth decrement.

To this end, schemes [1] [2] are proposed to dynamically adjust ATIM window size based on the length of idle time in both ATIM window period and data transmission period in previous beacon interval. However, existing work fails to provide a mechanism to setup initial ATIM window size but totally relies on the dynamic adaptation schemes to adjust ATIM window size to its optimal value. This is not only time consuming but also unstable in some cases when the traffic variation changes dramatically.

To this end, this work proposes new algorithms to calculate the optimal ATIM window size based on the number of nodes in the IBSS. This scheme allows system to dynamically decide the optimal ATIM window size when nodes join or leave the system. The scheme can work together with other dynamic adaptation schemes to get better performance.

In the next section we will present out main theoretical results. Then we will give the initial experimental verification together with some discussion on work in progress.

II. DERIVATION OF OPTIMAL ATIM WINDOW SIZE

A. Basic Assumption

Figure 1 shows the components of a beacon interval time T_{Total}. At every TBTT, average time for a possible data transmission from previous beacon interval to finish is T_R. Then every STA in the clique will compete to send out beacon frame. Time T_B is required for the first STA to successfully send out a beacon message and become the beacon station in current beacon interval. After hearing the beacon message, other STAs will cancel their beacon transmission and enter the phase to send out ATIM messages. This happens in the leftover time, T_A, within the ATIM window time, T_{ATIM}. Finally, nodes sending out ATIM/ACK message correctly will send data in the T_D period.

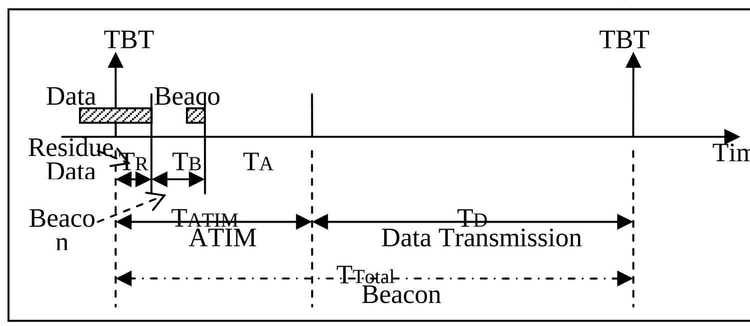


Figure 1 : Beacon interval time components

We also assume that all nodes in the given IBSS can hear each other and as a result form a clique. There is no priority difference among competing nodes. Every node that can transmit in the Data Transmission Period must have a corresponding successful ATIM/ACK exchange in the ATIM window. There is no inferred transmission between two nodes that have no direct ATIM/ACK exchange but can infer each other's power saving state.

We assume that the time interval between two adjacent transmission attempts is exponentially distributed. As a result, the channel attempt rate is assumed to follow a Poisson distribution with an average rate of λ. We also assume that the channel collision rate, p, is constant and only related to the current competing traffic load.

B. Derivation of average contention window size L, average channel attempt rate λ, and channel collision ratio p

Let N be the number of nodes in the neighborhood that are within each other's transmission range. The current average channel attempt rate, λ, can be represented by

$$\lambda = \sum_{i=1}^{N} \frac{1}{L_i} = \frac{N}{L} \qquad (1)$$

In Equ. 1, L_i is the average contention window of node i. We assume that every node in the same clique has the same average contention window size, namely, L.

Because the channel attempt rate is assumed to be Poisson distributed, the probability mass function in a time slot of k transmission is

$$\Pr[k] = \frac{\lambda^k}{k!} e^{-\lambda} \qquad (2)$$

Hence, channel collision ratio p is

$$p = \Pr[k \ge 2] = 1 - \Pr[0] - \Pr[1] = 1 - e^{-\lambda} - \lambda e^{-\lambda} \quad (3).$$

Let m be the number of retransmission that reaches maximum backoff window size. For the exponential backoff scheme, the probability that j^{th} collision window size occurs is

$$c_j = \begin{cases} c_0 p^j, & 1 \le j \le m-1, \\ c_0 \sum_{j=m}^{\infty} p^j = \dfrac{c_0 p^m}{1-p}, & j = m \end{cases} \qquad (4)$$

where $\sum_{j=0}^{\infty} c_j = 1$, and we also have

$$c_0 \cdot (1 + p + p^2 + \cdots + p^{m-1} + \frac{p^m}{1-p}) = 1 \Rightarrow c_0 = 1 - p \qquad (5) \cdot$$

Finally, let b_j be the jth contention window size. Then the average contention window size L is

$$L = \sum_{j=0}^{m} b_j \cdot c_j = \frac{CW_{\min}}{2(1-2p)}[1 - p - p(2p)^m] - \frac{1}{2} \quad (6).$$

Equations (1), (3) and (6) establish the relations among average contention window size L, average collision ratio p, and average channel attempt rate λ. $CW_j = 2^j \cdot CW_{\min}$ and $CW_{\max} = 2^m \cdot CW_{\min}$.

Given $L^{(0)}$, which represents the largest backoff window size, we can calculate $\lambda^{(0)}$ and $p^{(0)}$, respectively, using Equation (1) and (3). Then, by applying Equation (6), we can calculate $L^{(1)}$. The iteration repeats until the difference of two consecutive iteration values satisfies $|L^{(j+1)} - L^{(j)}| < \varepsilon$, where denotes some pre-defined small value. The iterative algorithm always converges as proved in Theorem 1 in [3]

C. Derivation of average number of STAs having successful ATIM/ACK exchange within ATIM window

Assume the average time used for ATIM message transmission is T_A. At first, we derive the number of successful transmissions, N_{f-suc}, that are achieved within T_A. Then, we derive the average number of STAs, N_{n-suc}, that have successful ATIM/ACK exchange within T_A and thus will keep awaken during data transmission period of a Beacon Interval. Note that N_{f-suc} in general is no less than N_{n-suc} because one STA might transmit several ATIM/ACK messages.

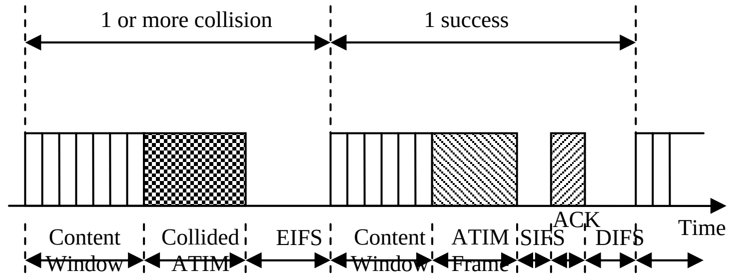


Figure 2: Sequence of one successful data frame transmission

To save space we only list main result below. The total time for a successful transmission, $t_{total-one}$, is

$$t_{total-one} = \frac{1-e^{-\lambda}}{\lambda e^{-\lambda}}(\frac{1}{\lambda} + t_{ATIM}) + \frac{1-e^{-\lambda} - \lambda e^{-\lambda}}{\lambda e^{-\lambda}} \cdot EIFS + 2SIFS + t_{ACK} + DIFS \; (7)$$

In Equ.7, EIFS, SIFS, and DIFS are predefined system specific values. t_{ATIM} and t_{ACK} are the time to transmit a ATIM frame and a ACK frame that can be calculated from packet length and channel transmission rate.

Then within T_A, the average number of total transmission, N_{f-suc}, is

$$N_{f-suc} = T_A / t_{total-one} \qquad (8) \quad .$$

Because some nodes may transmit several ATIM messages, we hereby calculate the total number of individual nodes, N_{n-suc}, that successfully send out ATIM/ACK messages. We assume that all N_{f-suc} frames are equally likely between any two nodes.

For each node i, a probability function, X_i, is defined as :

$$X_i = \begin{cases} 0 & node\ i\ is\ not\ among\ N_{f-suc}\ flows \\ 1 & node\ i\ is\ among\ N_{f-suc}\ flows \end{cases} \qquad (9)$$

Then $E[X_i]$ is the probability that node i is covered by one or several N_{f-suc} flows. Every node i has the same distribution.

N is the total number of nodes in the system. The expected number of individual nodes covered by N_{f-suc} flows is:

$$\begin{aligned} N_{n-suc} &= E[X_1 + X_2 + \cdots + X_N] = N \cdot E[X] \\ &= N\left[1 - \left(\frac{N-2}{N}\right)^{N_{f-suc}}\right] \qquad (10) \end{aligned} .$$

D. Derivation of average data transmission finishing time of all STAs within ATIM window

After ATIM window, only N_{n-suc} nodes keep awaken and compete to send out data frames. Assume that every node always has packets to send. We define the optimal length of data transmission period to be the time needed for every node to successfully transmit at least one packet.

Define Y_i to be the time needed for i^{th} node to successfully transmit first packet. Then the total time needed for all N_{n-suc} nodes to finish transmission is:

$$T_D = E[Y_1 + Y_2 + \cdots Y_{n-suc}] = E[Y_1] + \cdots + E[Y_{n-suc}] \qquad (11).$$

To calculate $E[Y_i]$, consider the probability for i^{th} node to send out a frame in a slot given there are $(i-1)$ nodes already transmit a frame.

$$\Pr[ith\ node\ transmits\ a\ \ frame] = \frac{N_{n-suc} - (i-1)}{N_{n-suc}} \qquad (12)$$

Because the required time is Geometric distributed,

$$N_D = \frac{N_{n-suc}}{N_{n-suc}} + \cdots + \frac{N_{n-suc}}{N_{n-suc} - (N_{n-suc} - 1)}$$

$$= N_{n-suc} \cdot (1 + \cdots + \frac{1}{N_{n-suc}}) = N_{n-suc}[\ln N_{n-suc} + o(1)] \qquad (13)$$

Equation (13) gives us the average successful transmission that is required for each node to transmit at least one data packet. In the data transmission period, the average time for one successful data transmission , $t_{data-one}$, can be derived in the same way as Eqa. (7) but using different channel attempt rate λ_d that is based on N_{n-suc} instead of N .

$$t_{data-one} = \frac{1-e^{-\lambda_d}}{\lambda_d e^{-\lambda_d}}(\frac{1}{\lambda_d} + t_{ATIM}) + \frac{1-e^{-\lambda_d} - \lambda e^{-\lambda_d}}{\lambda_d e^{-\lambda_d}} \cdot EIFS + 2SIFS + t_{ACK} + DIFS$$

Then, we have

$$T_D = N_D \cdot t_{data-one} \qquad (14)$$

E. Derivation of average beacon transmission time

The total time to successfully transmit a beacon, T_B, is

$$T_B = \frac{1-e^{-\lambda_B}}{\lambda_B e^{-\lambda_B}}(\frac{1}{\lambda_B} + t_{beacon}) + \frac{1-e^{-\lambda_B} - \lambda_B e^{-\lambda_B}}{\lambda_B e^{-\lambda_B}} \cdot EIFS + DIFS \quad (15)$$

F. Derivation of average beacon transmission time

The average time for a node to finish a packet transmission cross TBTT time boundary, T_R, is calculated as $t_{data}/2$. Let F_{data} be the average length of a data frame,

$$T_R = F_{data} / 2R_{trans} \qquad (16) \quad .$$

According to Figure 2, the total beacon interval, T_{Total}, is:

$$T_{Total} = T_R + T_B + T_A + T_D \qquad (17).$$

Finally, given T_{Total}, the optimal ATIM window size, T_{ATIM}, is:

$$T_{ATIM} = T_R + T_B + T_A \qquad (18).$$

III. PERFORMANCE EVALUATION

In this section we briefly describe the evaluation of our schemes. We have implemented a C++ based 802.11 IBSS power saving testbed and performed our ATIM optimization simulation in the testbed. All parameters are taken from 802.11 or from current popular configuration. Our analytical analysis is carried on in MATLAB. Some important parameters are: SIFS (0.01ms), slot time (0.02ms), Channel transmission rate (11Mbps), ATIM window size(30 ms), and beacon size (100 to 400 ms). The total number of nodes is between 5 and 25.

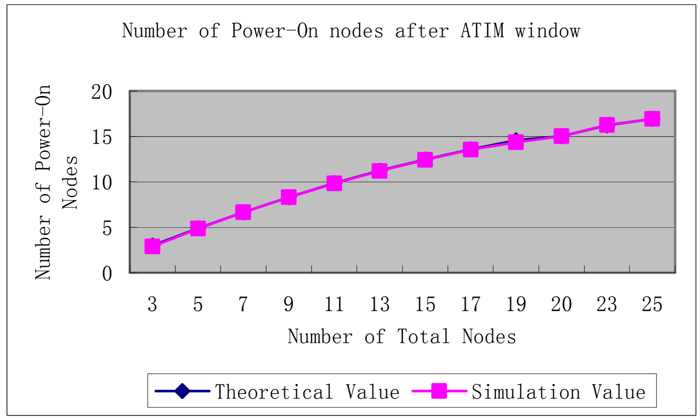
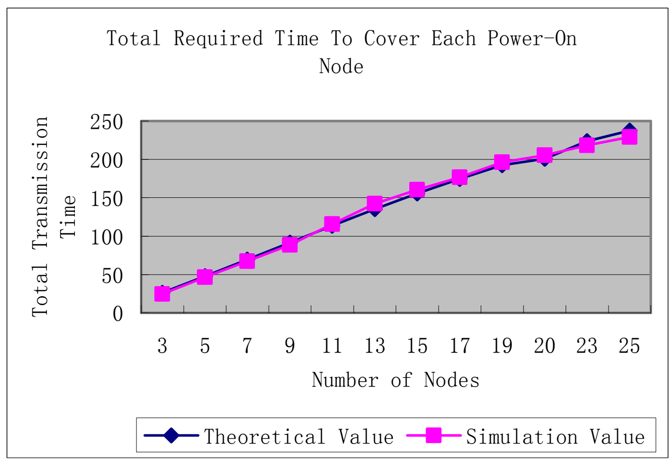


Figure 3: Number of power-on nodes N_{n-suc} after ATIM



F

Figure 4: T_D vs. the number of total nodes N

IV. CONCLUSIONS

Our initial experimental results have shown the accuracy of our analytical model. In most cases, the differences are very small. Currently we are investigating the performance when nodes are frequently join or leave the clique.

References

[1]. Z. Zhong, "Power management in an IEEE 802.11 IBSS WLAN using an adaptive ATIM window," US Patent application WO 2004/077762 A1, Sep. 2004.

[2]. Z. Zhong, "Power management in an IEEE 802.11 IBSS WLAN using an end of ATIM frame and a dynamically determine ATIM period," US Patent application WO04077763A1, Sep. 2004.

[3]. H. Kim, and J. Hou, "Improving protocol capacity with Model-based frame scheduling in 802.11-operated WLANs," Mobicom 2003, San Diego, Sep. 2003