Packet Delay Modeling of IEEE 802.11 Wireless LANs

P. Raptis[1], V. Vitsas[1], K. Paparrizos[2], P. Chatzimisios[3], A. C. Boucouvalas[3], P. Adamidis[1]

[1] Department of Information Technology,
Technological Educational Institution, Thessaloniki, Greece,
{praptis, vitsas}@it.teithe.gr

[2] Department of Applied Informatics,
University of Macedonia, Thessaloniki, Greece,
paparriz@uom.gr

[3] Multimedia Communications Research Group, School of Design, Engineering and Computing,
Bournemouth University, Fern Barrow, Poole, UK,
{pchatzimisios, tboucouv}@bournemouth.ac.uk

Abstract

This paper presents a new analytical model for calculation of the average packet delay of the IEEE 802.11 Distributed Coordination Function assuming ideal channel conditions. The new model applies to both basic and Request-to-Send / Clear-to-Send access modes and is based on calculating the average delay of packets successfully transmitted after a specific number of collisions and the corresponding probability. Results indicate that the proposed model is more accurate than a model proposed in the literature that calculates the same components. The accuracy of another model presented in the literature that evaluates the time delay at each backoff stage is studied.

1. Introduction

IEEE 802.11 is the most successful protocol utilized by Wireless Local Area Networks (WLANs). IEEE 802.11 specifications define Medium Access Control (MAC) and Physical Layer (PHY) sublayers for WLANs [1]. IEEE 802.11 MAC protocol is based on the carrier sense multiple access with collision avoidance (CSMA/CA) scheme and includes two modes for channel access i.e. the Distributed Coordination Function (DCF) and the Point Coordination Function (PCF). DCF specifies channel contention-based mode operation and PCF specifies channel access contention free mode. The DCF describes two techniques to transmit data packets; a two-way handshaking (DATA - ACK) called basic access and an optional four-way handshaking (RTS - CTS - DATA - ACK) called Request-To-Send/Clear-To-Send (RTS/CTS) access method.

There is a lot of research work in modeling the IEEE 802.11 DCF and studying performance metrics such as throughput [2][3][4][5], packet delay [6][7][8][9], packet drop probability [6] and packet drop time [6]. Bianchi [2] proposed a two-dimensional Markov chain model to calculate the saturated throughput of the DCF assuming that the channel is error free, that there are no hidden stations and that capture effect conditions are not present. Bianchi's model also assumes infinite packet retransmissions. Wu [5] improved Bianchi's Markov chain to calculate the throughput taking into account the packet's retransmission limit as specified in the standard. Chatzimisios et al. [6] employed Wu's Markov chain to develop a mathematical model that calculates additional performance metrics, namely the average packet delay, the packet drop probability and the average packet drop time. Vukovic [8] improved models of [2] and [5] by reducing these models from two-dimensional to one-dimensional Markov chains. The one-dimensional models are elegant and make simple computations. Vukovic also calculated the average packet delay for infinite and finite packet retransmissions by calculating the average delay of packets successfully received after a specific number of collisions and the corresponding probability.

In this paper we extend our work use Markov chain model of [5] to develop a new analytical packet delay model by calculating the same components as Vukovic. We also compare analytical results of our model to the models of Vukovic and Chatzimisios. Comparison results indicate that a) our and Chatzimisios models are more accurate than Vukovic and b) our and Vukovic models are more comprehensible than Chatzimisios as far as the components used for the calculation of the average delay is concerned.

2. Backoff Procedure

According to basic access if a station has a DATA packet to transmit and senses the channel to be idle for a period of Distributed Inter Frame Spacing (DIFS) then the station proceeds with its transmission. If the channel is busy, the station defers until an idle DIFS is detected and then generates a random backoff interval before transmitting in order to minimize collisions. The backoff time counter is decreased in terms of slot time as long as the channel is sensed idle. The counter is stopped when the channel is busy and resumed when the channel is sensed idle again for more than DIFS. A station transmits a packet when its backoff timer reaches zero. If the destination station successfully receives the packet, it waits for a short inter-frame space (SIFS) time interval and replies with an acknowledgement (ACK) packet. If the transmitting station does not receive an ACK packet within a specified ACK timeout interval, the data packet is assumed to have been lost and the station schedules a retransmission. Each station holds a retry counter that is increased by one each time a data packet is unsuccessfully transmitted. If the counter reaches the retransmission limit *m* the packet is discarded.

The backoff time counter is chosen uniformly in the range $[0, W_i\text{-}1]$, where *i* is the backoff stage $i \in [0, m]$ and W_i is the current contention window (*CW*) size. The contention window at the first transmission of a packet is set equal to $CW_{min}=W$. After an unsuccessful packet transmission the contention window *CW* is doubled up to a maximum value $CW_{max}=2^{m'}W$ (where *m'* is the number of *CW* sizes). Once *CW* reaches CW_{max} it remains in this value until it is reset. The *CW* is reset to CW_{min} after a successful packet transmission or if the packet's retransmission limit is reached.

The RTS/CTS access scheme follows the same backoff rules as basic access. When the backoff timer reaches zero, the station sends a short RTS packet first instead of the data packet. The receiving station responds with a CTS packet after a SIFS time interval. The sender is allowed to transmit the data packet only if it receives a valid CTS. Upon the successful reception of the data packet the receiver transmit an ACK frame.

3. Mathematical Modeling

This study assumes ideal channel conditions (no transmission errors or hidden stations), the contending stations are of fixed number *n* and each station has always a packet available for transmission of the same fixed size.

3.1. Markov Chain Model

Let *b(t)* and *s(t)* be the stochastic processes representing the backoff time counter and the backoff stage (0,…*m*) respectively for a given station at time *t*.

We utilize the same discrete-time Markov chain with [5][6] in order to model the two-dimensional process $\{b(t), s(t)\}$. The key approximation in this model is that each packet transmission collides with constant and independent probability *p* regardless of the backoff stage. Let $b_{i,k} = \lim_{t\to\infty} P\{s(t)=i, b(t)=k\}$ be the stationary distribution of the Markov chain, where $i\in[0,m]$, $k\in[0,W_i\ \text{-}1]$. The probability τ that a station transmits a packet in a randomly chosen slot time can be expressed as:

$$\tau=\begin{cases}\dfrac{2\cdot(1-2p)\cdot(1-p^{m+1})}{W\cdot(1-(2p)^{m+1})\cdot(1-p)+(1-2p)\cdot(1-p^{m+1})}, & m\le m' \\[2ex] \dfrac{2\cdot(1-2p)\cdot(1-p^{m+1})}{W\cdot(1-(2p)^{m'+1})\cdot(1-p)+(1-2p)\cdot[(1-p^{m+1})+W\cdot 2^{m'}\cdot p^{m'+1}\cdot(1-p^{m-m'})]}, & m>m'\end{cases} \quad (1)$$

The probability *p* that a transmitted packet encounters a collision is given by:

$$p = 1-(1-\tau)^{n-1} \quad (2)$$

Equations (1) and (2) represent a non-linear system with two unknown τ and *p*, which can be solved using numerical methods and has a unique solution.

3.2. Saturation Throughput

Let P_{tr} be the probability with that at least one station (out of *n*) transmits in a considered slot time:

$$P_{tr} = 1-(1-\tau)^n \quad (3)$$

Let P_s be the probability that a transmission occurring on the channel is successful and is given by the probability that only one station transmits and the *n*-1 remaining stations defer, with the condition that a transmission occurs on the channel. Probability P_s is given by:

$$P_s = \frac{n\cdot\tau\cdot(1-\tau)^{n-1}}{P_{tr}} = \frac{n\cdot\tau\cdot(1-\tau)^{n-1}}{1-(1-\tau)^n} \quad (4)$$

Let *E*[*slot*] be the average length of a slot time. *E*[*slot*] is given by:

$$E[slot] = (1-P_{tr})\sigma + P_{tr}P_sT_s + P_{tr}(1-P_s)T_c \quad (5)$$

where σ is the period of an empty slot. T_s and T_c are the time durations the channel is sensed busy during a successful transmission and a collision, respectively.

The saturation throughput *S* is defined as the fraction of time the channel is used to successfully transmit payload information:

$$S = \frac{P_{tr}\cdot P_s\cdot l}{E[slot]} = \frac{P_{tr}\cdot P_s\cdot l}{(1-P_{tr})\sigma + P_{tr}P_sT_s + P_{tr}(1-P_s)T_c} \quad (6)$$

where *l* is the packet length.

The time duration of T_s and T_c depends upon the channel access method employed. For the basic access method, we have:

$$T_s^{bas} = T_c^{bas} = DIFS+H+l+\delta+SIFS+ACK+\delta$$

and for the RTS/CTS access method:

$$T_s^{RTS} = DIFS+RTS+SIFS+\delta+CTS+SIFS+\delta+H+l +SIFS+\delta+ACK+\delta$$

$$T_c^{RTS} = DIFS+RTS+SIFS+CTS$$

where H is the packet header (equal to the sum of MAC and physical header) and δ is the propagation delay.

3.3. Average Packet Delay

The delay for a successfully transmitted packet is defined as the time interval from the time the packet is at the head-of-line of the queue ready to be transmitted, until an acknowledgement for this packet is received. If a packet reaches the specified retry limit then this packet is dropped and its time delay is not included in the calculation of the average packet delay.

3.3.1. Chatzimisios Model

A simple model was developed by Chatzimisios [6][9] for calculating the average packet delay $E[D]$ for both basic and RTS/CTS access mode. The average packet delay $E[D]$ is given by:

$$E_C[D] = \sum_{i=0}^{m} (E[X_i] \cdot k_i) \tag{7}$$

where X_i is the time period that a successfully transmitted packet delays at the i backoff stage and the k_i is the probability that a successfully transmitted packet reaches the i stage. $E[X_i]$ is given by:

$$E[X_i] = d_i \cdot E[slot] \qquad i \in [0, m] \tag{8}$$

where d_i is the average number of slot times the packet is delayed in the i backoff stage and $E[slot]$ is average slot time given from (5). The d_i is given by:

$$d_i = (W_i + 1)/2 \qquad i \in [0, m] \tag{9}$$

The probability k_i is given by:

$$k_i = \frac{p^i - p^{m+1}}{1 - p^{m+1}} \qquad i \in [0, m] \tag{10}$$

where (1- p^{m+1}) is the probability that the packet is not dropped and (p^i - p^{m+1})/(1- p^{m+1}) is the probability that a packet that is not dropped reaches the i stage.

From (7), (8), (9) and (10) we get Chatzimisios average packet dealy:

$$E_C[D] = E[slot] \cdot \sum_{i=0}^{m} \left(\frac{W_i + 1}{2} \cdot \frac{p^i - p^{m+1}}{1 - p^{m+1}} \right) \tag{11}$$

3.3.2. Vukovic Model

In [8] Vukovic proposed an analytical model that calculates the average packet delay based on average delay time of packets successfully received after a specific number of collisions and the corresponding probability. The average delay is calculated as the sum of time delays that a packet experiences a) deferring at all backoff stages, b) at packet's unsuccessful transmissions, and c) final packet's successful transmission. The average packet delay $E[D]$ is given by:

$$E_V[D] = \sum_{j=0}^{m} (E[V_j] \cdot q_j) \tag{12}$$

where V_j is the delay of a successfully transmitted packet from the j backoff stage and q_j is the probability that the packet is successfully transmitted from the j stage. The probability q_j (probability per stage) is given by:

$$q_j = \frac{p^j}{1 - p^{m+1}} \cdot (1 - p) \qquad j \in [0, m] \tag{13}$$

where (1- p) is the probability that a packet is successfully transmitted after the packet reached the j stage with probability p^j, provided that the packet is not dropped $(1 - p^{m+1})$.

The average delay $E[V_j]$ (delay per stage) is given by:

$$E[V_j] = T_s + j \cdot T_c + E[slot] \cdot \sum_{i=0}^{j} \left[\frac{W_i - 1}{2} \right] \text{ , } j \in [0, m] \tag{14}$$

where $(W_i$-1$)/2$ is the average number of slot times that the station defers in the i stage, $E[slot]$ is the average length of a slot time given from (5), jT_c is the time that the packet utilizes in collisions until it reaches the j stage, T_s is the time to transmit successfully from the jth stage.

From (12), (13) and (14) we get Vukovic's average packet delay:

$$E_V[D] = \sum_{j=0}^{m} \left(\left(T_s + jT_c + E[slot] \sum_{i=0}^{j} \frac{W_i - 1}{2} \right) \cdot \frac{p^j (1 - p)}{1 - p^{m+1}} \right) \tag{15}$$

3.3.3. New Model

Our model calculates the average packet delay using the same components as Vukovic model. The average packet delay $E[D]$ is given by:

$$E_N[D] = \sum_{j=0}^{m} \left(E[B_j] \cdot q_j \right) \tag{16}$$

where B_j is the delay of a successfully transmitted packet from the j backoff stage and q_j is the probability that the packet is successfully transmitted from the jth stage. The probability q_j calculated accurately by Vukovic and is given from (12). The delay B_j is calculated as the summary of the delays that a packet experiences at 0, 1,…, j stages. The average delay $E[B_j]$ is given by [10]:

$$E[B_j] = T_s + j \cdot T_c + E^{'}[slot] \cdot \sum_{i=0}^{j} \left[\frac{W_i - 1}{2} \right] \text{ , } j \in [0, m] \quad (17)$$

where $(W_i - 1)/2$ is the average number of slot times that the station defers in the ith stage, $E^{'}[slot]$ is the average length of a slot time when the remaining n-1 stations compete for the channel, jT_c is the time that the packet utilizes in collisions until reaches the j stage, T_s is the time to transmit successfully from the jth stage. $E^{'}[slot]$ is given by:

$$E^{'}[slot] = (1 - P_{tr}^{'}) \cdot \sigma + P_{tr}^{'} \cdot P_s^{'} \cdot T_s + P_{tr}^{'} \cdot (1 - P_s^{'}) \cdot T_c \quad (18)$$

where $P_{tr}^{'}$ is the probability with that at least one station out of n-1 transmits in the considered slot time and is given by:

$$P_{tr}^{'} = 1 - (1 - \tau)^{n-1} \quad (19)$$

and $P_s^{'}$ is the probability that a transmission occurring on the channel is successful and is given by the probability that only one station transmits of the n-1 remaining stations, with the condition that a transmission occurs on the channel:

$$P_s^{'} = \frac{(n-1) \cdot \tau \cdot (1-\tau)^{n-2}}{P_{tr}^{'}} = \frac{(n-1) \cdot \tau \cdot (1-\tau)^{n-2}}{1 - (1-\tau)^{n-1}} \quad (20)$$

From (16),(17) and (13) we get the new average packet delay:

$$E_N[D] = \sum_{j=0}^{m} \left(\left(T_s + jT_c + E^{'}[slot] \sum_{i=0}^{j} \frac{W_i - 1}{2} \right) \frac{p^j (1-p)}{1 - p^{m+1}} \right) \quad (21)$$

3.3.4. Remarks

In Chatzimisios model the probability component (10) includes at each stage the probabilities of successfully and unsuccessfully transmitted packets (term p^i in equation (10)), while in Vukovic and in the new model the probability component at each stage includes only the probability of successfully transmitted packets (term $p^i(1-p)$ in (13)). On the other hand Chatzimisios estimates the average backoff delay time during each transmission while the new model estimates the average delay time that each station uses to transmit its packets.

The delay component (14) of Vukovic model consists of three sub-components, a) the time T_s for packet's successful transmission, b) the time jT_c for j packet's unsuccessful transmissions, and c) the station's deferral time in each stage ($E[slot]$*the average number of slot times) i.e. the time that the station defers its transmission while the rest n stations compete for the medium. Note that the average slot time $E[slot]$ is computed for n stations. The difference of Vukovic model compared to our model is that we calculate the average slot time $E^{'}[slot]$ for n-1 stations since when a station defers does not compete for the channel. As result Vukovic model calculates the average packet delay for n+1 stations in a network of n stations or counts twice the activities of one station. In particular Vukovic uses the probabilities of equations (3) and (4) while we use equations (19) and (20).

3.4. Average Drop Time

A packet is dropped when it reaches the last backoff stage and experiences another collision. In [6][9], Chatzimisios proposed a simple mathematical model for estimation of average packet drop time:

$$E_C[T_{drop}] = E[slot] \cdot \sum_{i=0}^{m} \frac{W_i + 1}{2} \quad (22)$$

where $(W_i+1)/2$ is the number of slot times the packet is delayed in the i backoff stage and $E[slot]$ is average slot time given from (5).

From (17) we easily derive a new model for average drop time:

$$E_N[T_{drop}] = (m+1) \cdot T_c + E^{'}[slot] \cdot \sum_{i=0}^{m} \left[\frac{W_i - 1}{2} \right] \quad (23)$$

where $(W_i - 1)/2$ is the average number of slot times that the station defers in the ith stage, $E^{'}[slot]$ is the average length of a slot time when the remaining n-1 stations compete for the channel and $(m+1)T_c$ is the time that the packet utilizes in (m+1) collisions.

4. Results

The analytical results are compared to that taken from simulation outcome. The parameter values used for both simulation and analytical results follow the values specified for the Direct Spread Sequence Spectrum (DSSS) employed in the IEEE 802.11 [1] standard and are shown in Table 1.

Table 1. System Parameter Values

Channel bit rate, C	1 Mbit/s
Packet Payload	8184 bits
MAC header	224 bits
PHY header	192 bits
ACK	112 bits + PHY header
RTS	160 bits + PHY header
CTS	112bits + PHY header
Propagation delay, δ	1 μs
Slot time, σ	20 μs
SIFS	10 μs
DIFS	50 μs
Minimum CW, W_0	32
Number of CW sizes, m'	5
Short retry limit, m	6

Fig. 1 plots the average packet delay versus number of stations for packet size l=8184 bits and l=6000 bits for basic access. In this figure we compare analytical

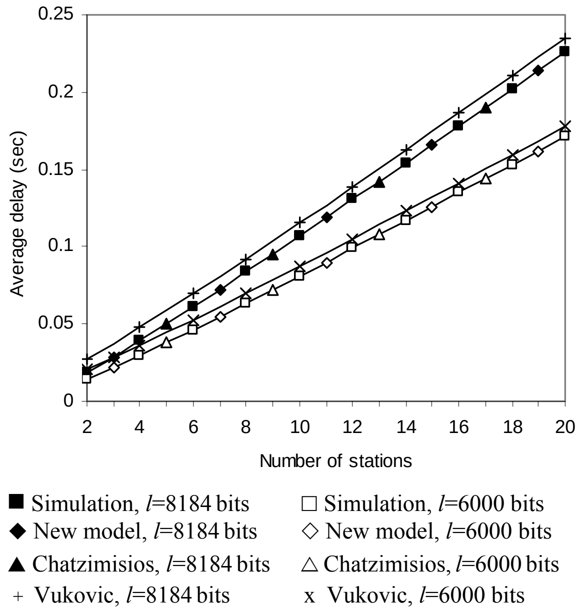

Fig. 1. Average packet delay versus number of stations for W=32, m=6, m'=5, C=1Mbit/sec and basic access.

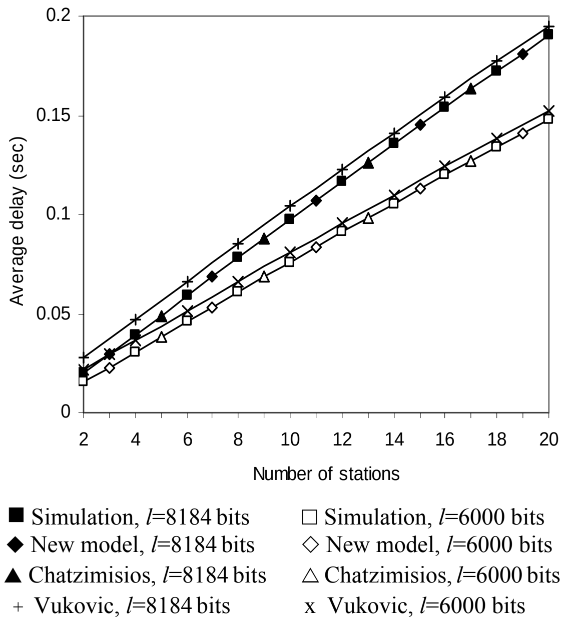

Fig. 2. Average packet delay versus number of stations for W=32, m=6, m'=5, C=1Mbit/s and RTS/CTS access.

results of the new model, Chatzimisios model, and Vukovic model to simulative results. The figure shows that the new and Chatzimisios models match exactly the simulation results. Vukovic model overestimates average packet delay for both packet sizes. This overestimation is high for small size networks (for n=2 is about 30%). The difference is getting lower as the network size increases (for n=20 is about 3%) while for large networks the difference is getting insignificant (for n=50 is about 1%) as the influence of the double counted activities of one station is negligible. Simulation results are taken with a 95% confidence interval lower than 0.006.

Fig. 2 plots the same results for the RTS/CTS access mode i.e. average packet delay versus number of stations for packet load l=8184 bits and l=6000 bits. The figure shows that the new and Chatzimisios models match exactly the simulation results while Vukovic model overestimates packet delay. The difference for n=2 is 30% (the same as in basic access) and for n=20 is 2% slightly smaller than basic access as the double counted station experiences more collisions that last less time.

In order to verify the source of overestimation we compare analytical results of the constituent parts of our model to the constituent parts of Vukovic model, namely the delay per stage ($E[V_j]$ opposed to $E[B_j]$) and the probability per stage (q_j same in both models). The analytical results are compared to simulation outcome.

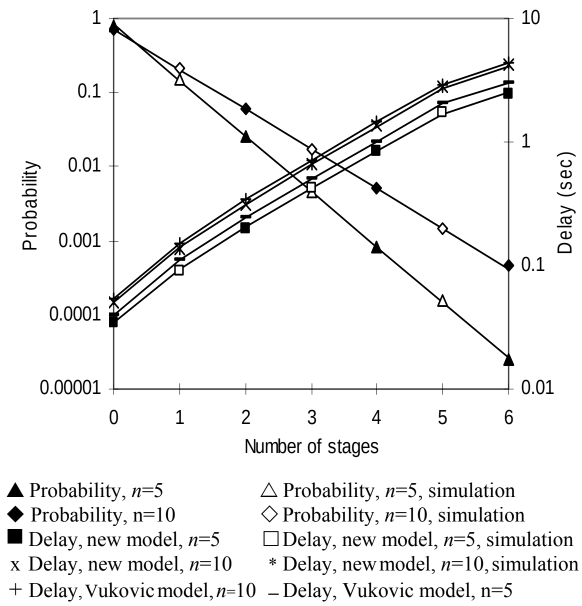

Fig. 3. Average packet delay per stage and probability per stage versus backoff stages for our model and Vukovic model, W=32, m=6, m'=5, C=1Mbit/s, n=5, n=10, l=8184 bits and basic access.

Fig. 3 plots the average packet delay per stage and the probability per stage respectively versus the number of backoff stages, for n=5 and n=10 and basic access mode. Fig. 3 shows that the delay per stage of the new model matches the simulation results. Vukovic model overestimates delay at all stages as the $E[slot]$ influences the delay at all stages. As expected, the difference is decreasing as the number of stations is increasing from n=5 to n=10.

Since constituent parts of Chatzimisios model are different from those of Vukovic and of the new model, we compare Chatzimisios model results only to simulation results. Fig. 4 plots the average delay of successful and unsuccessful packets at each stage ($E[X_i]$) and the probability use of stages (k_i) respectively versus the number of backoff stages, for n=10, basic and RTS/CTS access mode. The simulation results match exactly the analytical results at all stages verifying that Chatzimisios model accurately calculates the average packet delay. As expected the delay for RTS/CTS mode is smaller than for basic basic access. The delays at stage 5 and stage 6 are the same as these stages use the same contention window size (CW=1024). The probability at stage 6 includes only the probability of successfully transmitted packets.

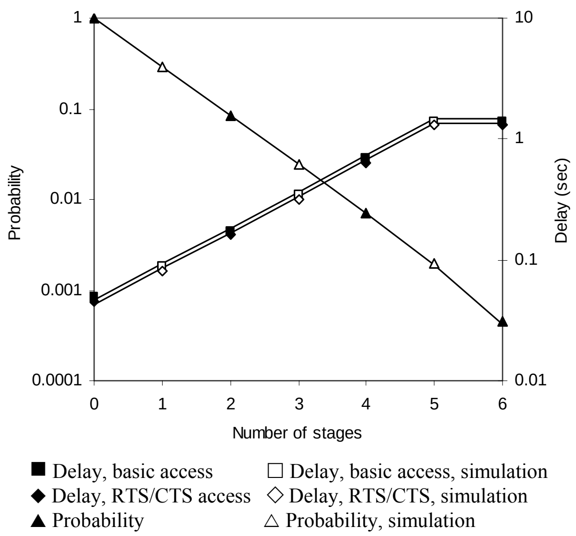

Fig. 4. Packet delay and probability versus backoff stages for Chatzimisios model, W=32, m=6, m'=5, C=1Mbit/s, n=10, l=8184 bits and both basic and RTS/CTS access.

5. Conclusions

This paper proposes a new analytical model to compute the average packet delay of 802.11 protocol. The average packet delay is calculated as the sum of a) the time that a station defers before transmitting the packet, b) the time that a station utilizes in packet collisions, and c) the time that the packet is successfully transmitted. Results of the new model are compared to the results of two models proposed in the literature i.e. Vukovic and Chatzimisios. Performance results show that our model is more accurate compared to Vukovic model and that our model uses more comprehensible components in calculating average packet delay than Chatzimisios model. The comparison is supported by simulation results.

Acknowledgements

This work is funded by the Greek Ministry of Education (25%) and European Union (75%) under the EPEAEK II program "Archimedes".

6. References

[1] IEEE *standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, IEEE Std: 802.11b-1999/Cor 1-2001, 2001.

[2] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function", *IEEE Journal on Selected Area in Communications*, vol.18, no.3, pp. 535-547, 2000.

[3] H.S. Chhaya and S. Gupta, "Performance Modelling Of Asynchronous Data Transfer Methods Of IEEE 802.11 MAC Protocol", *Wireless Networks*, vol. 3, no. 3, pp. 217-234, 1997.

[4] F. Cali, M.Conti, and E.Gregori, "Dynamic Tuning of the IEEE 802.11 Protocol to Achieve a Theoretical Throughput Limit", *IEEE Transactions on Networking*, vol. 8, no. 6, pp. 785-790, December 2000.

[5] H. Wu, Y. Peng, K. Long, S. Cheng, J. Ma, "Performance of Reliable Transport Protocol over IEEE 802.11 Wireless LAN: Analysis and Enhancement", Proc. of *IEEE INFOCOM*, pp.599-607, June 2002.

[6] P. Chatzimisios, A. C. Boucouvalas and V.Vitsas, "IEEE 802.11 Packet Delay – A Finite Retry Limit Analysis", *IEEE GLOBECOM,* vol. 2, pp.950-954, December 2003.

[7] Z. Hadzi-Velkov and B. Spasenovski, "Saturation Throughput-Delay Analysis of IEEE 802.11 DCF in Fading Channel", Proc. of *IEEE International Conference on Communications (ICC)*, vol. 1, pp. 121-126, May 2003.

[8] I. Vukovic and N. Smavatkul, "Delay Analysis of Different Backoff Algorithms in IEEE 802.11", Proc. of *IEEE Vehicular Technology Conference* (*VTC)*, Los Angeles CA, September 2004.

[9] P. Chatzimisios, "Performance Modelling and Enhancement of Wireless Communication Protocols", *PhD thesis*, School of Design, Engineering & Computing (DEC), Bournemouth University, U.K, Dec. 2004.

[10]P. Raptis, V. Vitsas, K. Paparrizos, P. Chatzimisios and A. C. Boucouvalas, "Packet Delay Distribution of the IEEE 802.11 Distributed Coordination Function", Proc. of *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WOWMOM 2005)*, Taormina, Italy, June 2005.