Neighbour -Aware, Collision Avoidance MAC Protocol (NCMac) for Mobile Ad Hoc Networks

Sylwia Romaszko
University of Antwerp - IBBT
PATS Research Group
Middelheimlaan 1, B-2020 Antwerpen
Email: sylwia.romaszko@ua.ac.be

Chris Blondia
University of Antwerp - IBBT
PATS Research Group
Middelheimlaan 1, B-2020 Antwerpen
Email: chris.blondia@ua.ac.be

Abstract— **IEEE 802.11 [1] has become the standard for Wireless LAN's both in infrastructure and in ad hoc mode. The standard specifies the medium access control mechanism, Distributed Coordination Function (DCF). Deploying the 802.11 DCF in wireless multi-hop networks leads to inefficient utilization of bandwidth and energy due to many unnecessary collisions. To attenuate these problems we propose a Neighbour -aware Collision avoidance MAC protocol (*NCMac*). The NCMac protocol defines algorithms to estimate minimum and maximum Contention Window (*CW*) sizes (CW_{min}, CW_{max}) taking into account the number of neighbours in the one-hop neighbourhood and the energy level of the battery. A novel resetting algorithm applied both after a successful and unsuccessful transmission has been also designed. In [3] we have already proposed a similar algorithm to calculate the CW_{min} (but not CW_{max}). However, the resetting algorithm of the NCMac is enhanced and unified and the NCMac does not modify the backoff algorithm defined in [1]. The NCMac protocol achieves a better throughput and noticeably reduces the number of collisions resulting in a longer network lifetime and a later dead of the first node as compared with the IEEE 802.11 DCF standard.**

I. INTRODUCTION

The 802.11 DCF is based on CSMA/CA -Carrier Sense Multiple Access with Collision Avoidance. The nature of the infrastructure environment is far from the real ad hoc conditions. The IEEE 802.11 specifies the *Binary Exponential Backoff (BEB)* algorithm, where the duration of a backoff period is selected randomly in a range limited by 0 and a certain CW value. After each collision the CW value is doubled. Upon a successful transmission or dropped packet the CW value is reset to the minimum. This solution is not only unfair but also inefficient. When the number of active neighbours increases, the number of collisions increases as well. Receiving packets successfully does not mean that the contention level has been dropped, only that the terminal has chosen a convenient CW value. In the 802.11 DCF standard the CW_{min} and CW_{max} are fixed, thus the network load is also not taken into consideration.

In this work we focus on dynamically adjusting the minimum and maximum value of the contention window interval and dynamic resetting of the CW value after successful transmission and dropped packet. The aim of a dynamic selection of the CW_{min}, CW_{max} and CW resetting is to decrease the probability of collisions and prolonging the lifetime of the network (last active node died) and a later dead of the first node.

The remainder of the paper is organized as follows. In Section II the IEEE 802.11 DCF standard is overviewed. In Section III the related work is presented. In Section IV the proposed protocol is described. In Section V we provided simulation results and analysis. Finally, concluding remarks are formulated in the last section.

II. OVERVIEW OF THE IEEE 802.11 DCF

IEEE 802.11 defines four kinds of *interframe space (IFS)* periods of time between frames: Short IFS, PCF IFS, DIFS (DCF IFS) and Extended IFS. They determine the priority level for accessing the channel. The DIFS duration is used by a station operating under the DCF mechanism. During this period a station senses the medium and starts the frame exchange if the channel has been sensed as idle and its *backoff time* has expired. The backoff time is an additional random deferral time before transmitting measured in slot time, $Backoff\ Time = Random() * Slot\ Time$. The *Random()* is a pseudorandom integer from the uniform distribution over the interval *[0,CW (Contention Window)]*. In the DCF 802.11 the minimum and maximum *CW* size (CW_{min}, CW_{max}) are fixed, where $CW_{min} <= CW <= CW_{max}$. The backoff interval initializes the backoff timer. It is decreased when the medium is idle and it is frozen when the medium is busy. When the backoff timer expires, the node can transmit. Upon a successful transmission, the *CW* is set to the minimum. When a collision occurs an exponential backoff mechanism is used. According to this mechanism, the *CW* is doubled after each unsuccessful transmission ($CW_{NEW} = 2 * (CW_{OLD} + 1)$) and nodes execute a new backoff process. Once it reaches the maximum *CW* size the *CW* remains the maximum value until it is reset to the minimum (CW_{min}) as depicted in Fig. 1.

III. RELATED WORK

In recent years many approaches enhancing the IEEE 802.11 standard have been proposed to reduce the number of collisions. The main concept of the Double Increment Double Decrement (DIDD) algorithm, introduced in [12], is decreasing the CW gently and gradually after a successful packet transmission. In case of a successful packet transmission, the
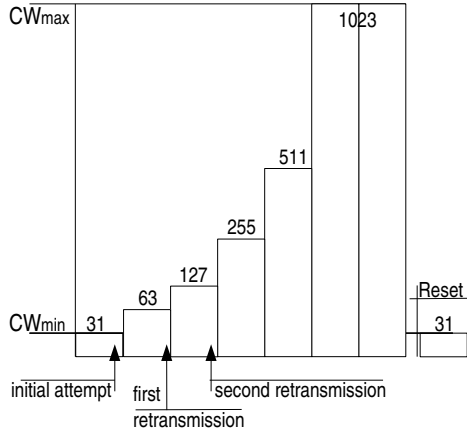
Fig. 1. *BEB* algorithm

Energy level	100%	85%	65%	45%	25%
RE	0.55	0.65	0.75	0.80	0.85
$Nr_{NEIGH} = 4$	26	31	37	41	43
$Nr_{NEIGH} = 8$	57	68	80	85	90

DIDD halves the CW instead of going back to CW_{min} in order to avoid potential future packet collisions. The Fast Collision Resolution MAC protocol has been described in [7]. The protocol allows the latest successful nodes to use a smaller CW and some nodes can reduce their backoff timer exponentially when they continuously meet idle time slots. In [8], the authors have shown that the number of contending nodes strongly affects the optimal CW size and they have proposed the Multiplicative Increase and Linear Decrease algorithm. However, the throughput degradation is observed when the number of nodes is large and the number of active nodes changes sharply from high to low. In [9], a similar scheme has been designed, called Linear/Multiplicative Increase and Linear Decrease. It uses an additional piece of information, namely, the overheard collisions. In [6], the authors have claimed that they had found the minimum optimal CW size which is equal to $8.5 * N - 5$, where N is the number of active nodes. Although the CW_{min} is estimated dynamically in [6], the CW_{max} is assumed to be constant and equal to 1023 slots like in [1]. In [5], the proposed collision resolution scheme allows the CW to adjust dynamically incorporating the knowledge of the collision process via the standard mean and variance statistics. In [4], an energy -aware Mac layer enhancement for IEEE DCF 802.11 dynamically tunes the backoff timer for fresh and collided data packets based on the node's energy. It changes the deferring time before transmitting fresh data packets and also after an unsuccessful transmission attempt based on a normal distribution with mean and variance that depend on the current node's battery level. In [11], a modified backoff mechanism uses a logarithmic increment instead of an exponential extension of the CW size in order to eliminate the degrading effect of a random number distribution.

IV. PROPOSED PROTOCOL

A. Calculating CW_{min} and CW_{max}

The CW depends on the number of failed retransmission attempts that are usually caused by collisions. The value of CW should be influenced by an element which mostly affects the number of collisions, such as the current number of neighbours in the 1-hop neighbourhood (Nr_{NEIGH}). An increase in the number of collisions degrades the throughput performance. It causes not only a high delay but also higher energy consumption, because the nodes try to retransmit, until the number of retransmission attempts reaches the maximum. Therefore the CW value should take the energy level of the battery into consideration. Towards this goal, our CW_{min} size is estimated based on the number of nodes in the 1-hop neighbourhood (Nr_{NEIGH}) and a coefficient of the remained energy (RE). Each node can estimate how many neighbours it has in its 1-hop neighbourhood, based on successfully detected signals or using the table that is built by the routing mechanism. The CW_{min} and CW_{max} values are dynamically calculated according to the following formulas:

$$CW_{min} = \alpha * Nr_{NEIGH} * RE - \beta; \qquad (1)$$

$$CW_{max} = \alpha * Nr_{NEIGH} * RE - \beta + \theta; \qquad (2)$$

where α and β are variables, that can be optimally tuned depending on the environment and θ is a constant and equals 2^{10}. Based on our extensive simulations [3] we have found that optimal values α and β are 14.0 and 5.0 respectively. The coefficient of the remained energy considers five battery levels (see Table I). Depending on a particular energy level the RE value varies. Notice that the value of RE increases, if the energy level decreases. Nodes with a lower battery level wait longer in order to avoid an eventual collision as much as possible. If a node has the energy level equal to or lower than 45% of its initial energy, the RE value increases slower, because the value of the CW is already large.

B. The resetting algorithm

According to the standard, upon a successful transmission or maximum transmission attempts, the CW is reset to the minimum value. However, receiving a packet successfully does not mean that the contention level is dropped. In case of a dropped packet this assumption is even more doubtful. The *resetting algorithm* has been designed to avoid unnecessary collisions in these cases. The value of CW is reset based on the number of 1-hop neighbours, their change during the recovery mechanism, the number of retransmission attempts. The influence of the number of 1-hop neighbours plays an important role for the network performance. Therefore, we have defined the variable NrN_{THR}, which is the established maximum number of 1-hop active neighbours. We have first assumed that the NrN_{THR} is fixed and equal to 6 in most of the simulations. However, this value can be
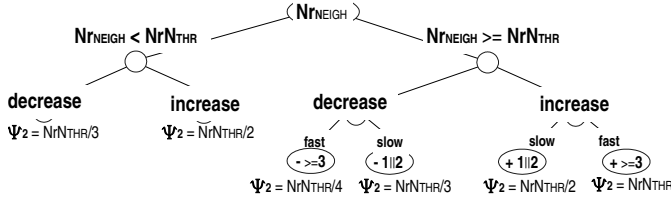
Fig. 2. Resetting algorithm

modified taking into account instantaneous channel conditions (e.g., SINR) or higher data rates. We have also defined a concept of the fast/slow increase (decrease) of the number of active 1-hop neighbours. We assume that our threshold FS_thresh equals $\frac{NrN_{THR}}{2}$. If $Nr_{NEIGH} => NrN_{ATT1} + FS\_thresh$ $(Nr_{NEIGH} <= NrN_{ATT1} - FS\_thresh)$ the fast increase (decrease) occurs, and the slow otherwise (where the NrN_{ATT1} is the number of neighbours when the first retransmission occurs).

First we reset the *Contention Window* value according formula (1), then an extra ψ value is added, thus $CW_{min} = CW_{min} + \psi$. The ψ value consists of ψ_1 and ψ_2. The ψ_1 is estimated according following formula:

$$\psi_1 = Nr_{NEIGH} * (1 - CW_{min}/CW_{old}) * \chi \qquad (3)$$

where CW_{old} is the last value of the CW after which a packet is received. We consider following cases during selection of the χ value:

1) When a packet has been received successfully and the $nr_{ATT} == 0$ then $\chi = 0$, thus $\psi_1 = 0$.
2) When a packet has been received successfully and $nr_{ATT} > 0$ then $\chi = \frac{nr_{ATT}+1}{10}$.
3) Finally, when the packet has been dropped $(nr_{ATT} > limit)$ then $\chi = 1$.

The ψ_2 value is reached as depicted in Fig. 2. Surveying the figure we see, that first the current number of 1-hop neighbours (Nr_{NEIGH}) is compared with a threshold of the number of neighbours (NrN_{THR}). We have assumed that NrN_{THR} equals 6 in this figure. Secondly, if Nr_{NEIGH} is smaller than the threshold, we check if an increase/decrease of the number of 1-hop neighbours has occurred comparing the Nr_{NEIGH} with the number of neighbours when first retransmission has occurred (NrN_{ATT1}). If the Nr_{NEIGH} is larger or equal to the threshold, we check if the fast/slow increase/decrease has occurred. Depending on the case a certain value of ψ_2 is added. Summarizing, the ψ value depends on:

1) the number of retransmission attempts (nr_{ATT})
   a) If *(no retransmissions)* only the change of the numbers of 1-hop neighbours is taken into account
   b) If*(retransmissions <= max)* both ψ_1 and ψ_2 are important
   c) If *(pkt dropped)* then $\psi = (\psi_1 + \psi_2)$ and $\chi = 1$
2) the increase/decrease (fast/slow) of the number of neighbours ψ_2. If *(no increase/decrease)* occurs then $\psi_2 = 0$;

V. PERFORMANCE EVALUATION

A. Metrics and parameters

The proposed NCMac protocol has been implemented in the ns-2 network simulator [2]. The simulations have been carried out for various topologies, scenarios with different kinds of traffic and routing protocols. The following performance metrics are used:

- Number of collisions
- Total packets received
- FND - Fist Node Died
- LND - Last active Node Died
- Average aggregate delay

We have defined the network lifetime as the time duration from the beginning of the simulation until the instant when the *active* (a node transmitting/receiving) Last Node Died, i.e. there is no live transmitter-receiver pair left in the network. In Table II we present the general simulation parameters. Other changed parameters in particular simulations are mentioned in corresponding paragraphs.

Mobility affects the MAC protocols because the set of users competing for capacity on the medium keeps changing. This makes it difficult to allocate bandwidth in an equitable fashion. In reality in mobile ad hoc networks the environment is also changing continuously. Therefore, in all simulations we have applied the *Shadowing Propagation Model* [2] with different parameters in order to analyze performance of our protocol and the IEEE DCF 802.11 standard in different interference environments, so with different fading effects.

The energy model supported by ns-2 [2] includes transmitting (tx), receiving (rx), idle and sleep state. Every node has an

initial value corresponding to the node energy level at the beginning of the simulation. A node consumes energy as it transmits and receives data. It also updates the amount of the energy spent in idle state. The interface has very large idle energy consumption when it operates in ad hoc mode, as shown in [10]. Therefore, we have assumed that the receive power ($rxPower$) is approximately 45% of the maximum transmit power ($txPower$) and the idle power ($idlePower$) is approximately 30% of the maximum transmit power.

B. Simulation and results

Fig. 3, 4 and 5 show the performance of the standard (STD) [1], CW algorithms (CW_{min_max}) applied to the standard and *NCMac* with 50 mobile nodes in 15 scenarios (other parameters can be found in Table II) in heterogeneous networks. Each flow has a different data rate. Both the NCMac with and without the *resetting algorithm* significantly outperforms the standard. Notice that in some scenarios (e.g. 2, 8, 10, 12) the total packets received by the 802.11 mechanism is much lower than the NCMac and CW_{min_max}. In the worst case scenario (5th scenario) the CW_{min_max} still receives 30 data packets more than the standard, and the NCMac 80 packets more. If we look at Fig. 4 and 5 we can observe that in this case both enhancements also achieve better performance of the LND and FND. The NCMac increased the total of received data packets by 33.49%, the lifetime of the network by 21.13% and the FND by 21.21%.
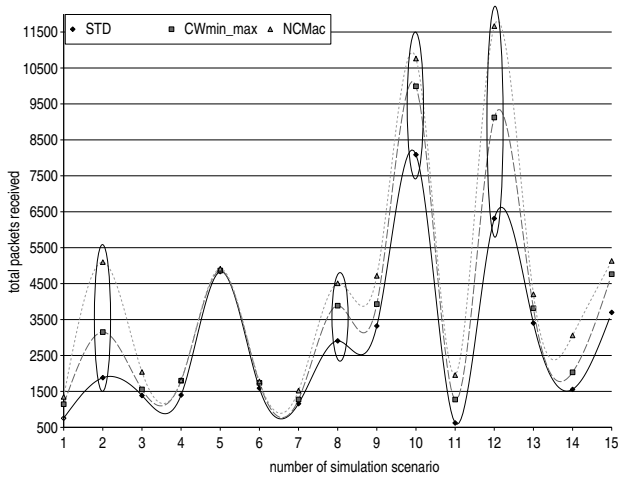

Fig. 3.   Total packets received

We have also executed a simulation with 25 nodes in a 500x500 meters shadowed urban area in 55 scenarios. In this simulation we wanted to analyze the behavior/sensitivity of the NCMac on the movement speed of nodes and the change (slow and fast) of the 1-hop neighbours. We realize that errors in the evaluation of the number of neighbours can occur, especially when nodes move fast. Therefore, in order to analyze the robustness to such errors, we have run 5 different topologies with 11 various maximum speeds of nodes ($0.03, 0.25, 0.5, 1, 1.5, 2, 2.5, 2.75, 3, 4, 5$) in the homogeneous
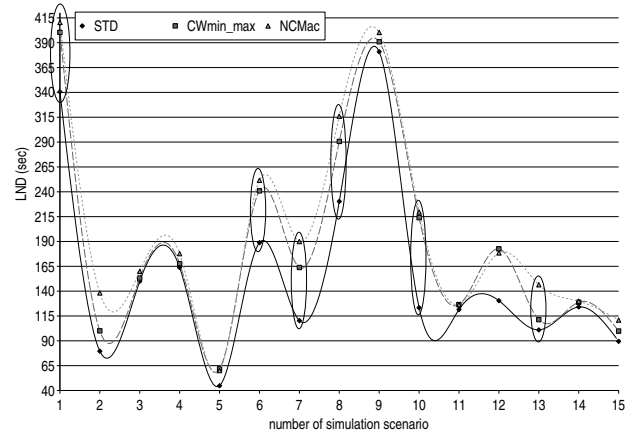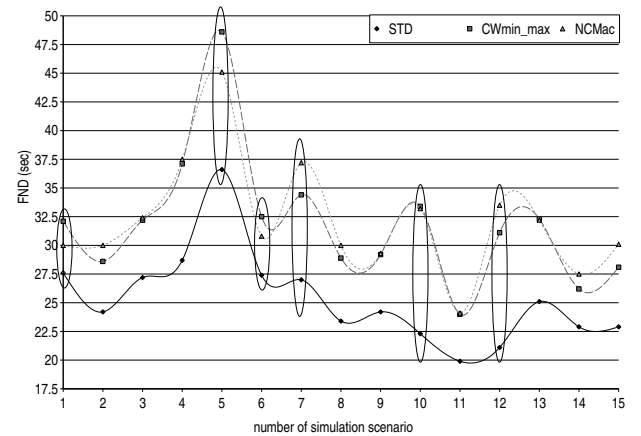

Fig. 4.   Lifetime of the network

Fig. 5.   FND

network. The RTS/CTS scheme has been used. In Table III we present the simulation parameters (other parameters remain like in Table II). Fig. 6 and 7 show the average aggregate First Node Died, average aggregate total packets received versus maximum speed of nodes (m/s) of the standard (STD) [1], and $NCMac$ protocols, respectively. We have observed that in both the standard and NCMac protocol, the total number of data packets decreases when the speed of the nodes increases. However this decrease is recompensed by a later dead of the first node as we can see in Fig. 6. This behavior is explicable. Since a node moves faster it has a worse connectivity with its recipient so less transmissions are executed. It can move towards its recipient, but it can also have more interferences from other terminals or environment's obstacles on the way. Notice that with an increasing speed of nodes the NCMac protocol achieves an increasing FND performance and a decreasing throughput performance over the DCF standard. In fast movement environment, if a node thinks that it has more neighbours than it has in real, it is more careful (it defers longer) avoiding more collisions and saving more energy. The NCMac protocol also takes into account the energy level of the battery when estimating the CW_{min}

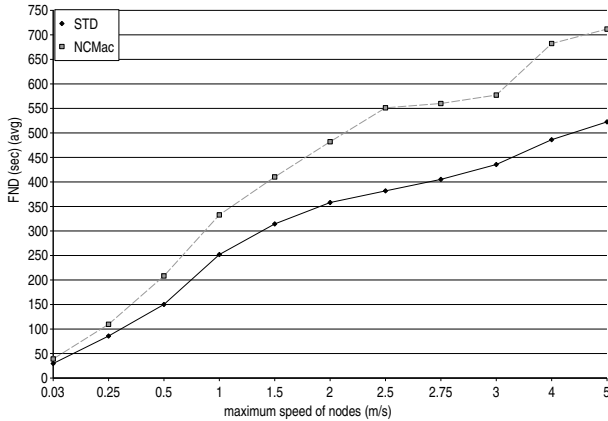| Parameter	Values
Number of active nodes	25
Simulations area (m)	500x500
Initial Energy (J)	5.0
Radio Propagation Model	Shadowed Urban Area
Payload size (bytes)	512, 1280, 2304
Nr of simulation scenarios	56
Maximal speed (m/s)	$0.03 - 5$

Fig. 6. Average FND (sec) versus maximum speed of nodes (m/s)

Fig. 7. Average aggregate total packets received versus maximum speed of nodes (m/s)

1-hop neighbours and the energy level of battery. The NCMac protocol achieves better throughput and noticeably reduces the number of collisions resulting in longer network lifetime and later dead of the first node as compared with the 802.11 DCF standard.

and CW_{max} values, which is an important issue in mobile ad hoc networks. The $NCMac$ outperforms the DCF standard by increasing the total of received data packets by 23.89%, the lifetime of the network by 25.93% and decreasing the number of collisions by 48.11%. Although the increasing speed of nodes and possible miscalculations of the number of nodes in 1-hop neighbourhood, the NCMac still outperforms the standard, because it can still profit from more fairly adjusted values of the CW_{min} and CW_{max} taking into account network load. The resetting algorithm also plays here important role by considering movement of the nodes, their fast or slow increase.

In our future work we will show a comparison the NCMac with the NEWCAMac from our previous work [3] in homogeneous and heterogenous networks with and without a novel backoff algorithm designed in [3]. Based on prelimary results in homogeneous networks with the basic access scheme, we have noticed than the NCMac outperforms the NEWCAMac (without a novel backoff algorithm) 7.63% in terms of the throughput and 11.36% in terms of the FND.

VI. CONCLUSION

We have designed a new MAC protocol with a novel resetting algorithm based on the number of neighbours in the 1-hop neighbourhood, a coefficient of the fast/slow increase/decrease of 1-hop neighbours and the number of re-transmission attempts. We select the minimum and maximum Contention Window sizes taking into account the number of
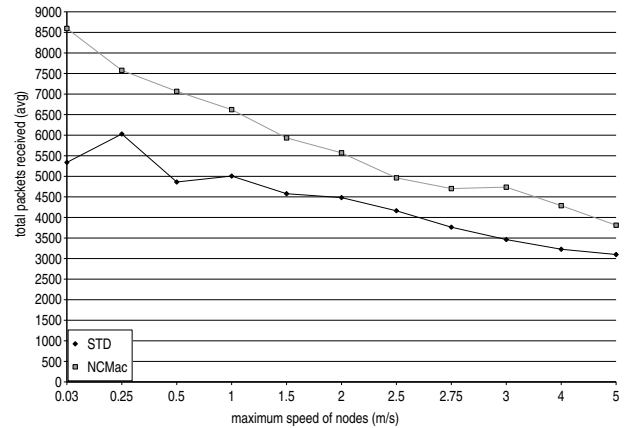
REFERENCES

[1] IEEE Computer Society LAN MAN Standards Committee: *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.* ANSI/IEEE Std. 802.11, 1999 Edition, The Institute of Electrical and Electronic Engineers, New York, 1999
[2] The Network Simulator NS: http://www.isi.edu/nsnam/ns
[3] S. Romaszko and C Blondia: *Neighbour and energy -aware contention avoidance MAC protocol for wireless ad hoc networks*, to appear in Proc. WiMob2006, June, Canada, 2006
[4] S.Gobriel, R.Melhem and D.Mosse, *BLAM: An Energy -Aware MAC Layer Enhancement for Wireless AdHoc Networks*, in Proc. WCNC04, Atlanta, USA, 2004
[5] J.Hu, C.D.Raymond, *A statistics based design of MAC protocols with distributed collision resolution for Ad Hoc networks*,in Proc. Mobi-Wac05, Hawaii, June, 2005
[6] M.Taifour, F. Nait-Abdesselam, D.Simplot-Ryl, *Neighbourhood backoff algorithm for optimizing bandwidth in single hop wireless ad-hoc networks*, in Proc. MobiWac05, Hawaii, June, 2005
[7] Y.Kwon,Y.Fang, and H.Latchman, *A novel MAC protocol with fast collision resolution for wireless LANs*, in Proc. IEEE INFOCOM, Apr. 2003
[8] V.Bharghavan, A.Demers,S.Shenker, and L.Zhang, *MACAW: a media access protocol for wireless LANs", in Proc. of the conference on communication archtectures, protocols and applications.* 1994, pp. 215-225,ACM Press
[9] J.Deng, P.K.Varshney, and Z.J.Haas, *A new backoff algorithm for the IEEE 802.11 Distributed Coordination Function*, in Proc. CNDS 04, San Diego, CA, USA, Jan.18-21,2004
[10] L. M. Feeney, *Investigating the energy consumption of an IEEE 802.11 network interface*, SICS Technical Report, T99/11, Dec. 1999
[11] S.S.Manaseer, M. Ould-Khaoua, "Logarithmic based backoff algorithm for MAC protocol in MANETs", Technical Report, University of Glasgow, http://www.dcs.gla.ac.uk/publications/, 2006
[12] P. Chatzimisios, A. C. Boucouvalas, V. Vitsas, A. Vafiadis, A. Economidis, P. Huang,"A simple and effective backoff scheme for the IEEE 802.11 MAC protocol", invited paper, in Proc. CITSA'05, Orlando, July 2005