# MINIMIZING MOBILES COMMUNICATION TIME USING MODIFIED BINARY EXPONENTIAL BACKOFF ALGORITHM

Ibrahim Sayed Ahmad[1], Ali Kalakech[1], Seifedine Kadry[2]

[1]Arts, Sciences and Technologies University, Lebanon

[2]American University of the Middle East, Kuwait

***ABSTRACT***

*The domain of wireless Local Area Networks (WLANs) is growing speedily as a consequence of developments in digital communications technology. The early adopters of this technology have mainly been vertical application that places a premium on the mobility offered by such systems. Examples of these types of applications consist of stocking control in depot environments, point of sale terminals, and rental car check-in. Furthermore to the mobility that becomes possible with wireless LANs; these systems have also been used in environments where cable installation is expensive or impractical. Such environments include manufacturing floors, trading floors on stock exchanges, conventions and trade shows, and historic buildings. With the increasing propagation of wireless LANs comes the need for standardization so as to allow interoperability for an increasingly mobile workforce. Despite all the advantages and facilities that Wi-FI offers, there is still the delay problem that is due to many reasons that are introduced in details in our case study which also presents the solutions and simulation that can reduce this delay for better performance of the wireless networks.*

*Binary Exponential Backoff (BEB) refers to a collision resolution mechanism used in random access MAC protocols. This algorithm is used in Ethernet (IEEE 802.3) wired LANs. In Ethernet networks, this algorithm is commonly used to schedule retransmissions after collisions.*

*The paper's goal is to minimize the time transmission cycle of the information between mobiles moving in a Wi-Fi by changing the BEB algorithm. The Protocol CSMA / CA manage access to the radio channel by performing an arbitration based on time. This causes many problems in relation to time transmission between mobiles moving in a cell 802.11. what we have done show that the protocol using CSMA / CA access time believed rapidly when the number of stations and / or the network load increases or other circumstances affects the network.*

***KEYWORDS***

*Network Protocols, Wireless Network, Binary Exponential Backoff.*

## 1. INTRODUCTION

With wireless networking, you don't need cables to connect your mobile devices. Instead, wireless networks use radio waves to send and receive network signals. As a result, a mobile device can connect to a wireless network at any location in your office.

Wireless networks are especially useful for laptops, smartphones, and tablets. After all, the main benefit of these devices is you can carry them around with you wherever you go. For example, at work, you can use your laptop at your desk, in the conference room, in the break room, or even out in the parking lot. With wireless networking, your portable devices can be connected to the network no matter where you take it.

A wireless network is a network that uses radio signals rather than direct cable connections to exchange information. An example of this kind of network includes devices from Dell's PowerConnect W-Series, powered by Aruba.

A computer with a wireless network connection is like a cellphone. Just as you don't have to be connected to a phone line to use a cell phone, you don't have to be connected to a network cable to use a wireless mobile device.

## 2. BINARY EXPONENTIAL BACKOFF

Classic Ethernet uses the 1-persistent CSMA/CD algorithm, this descriptor just means that stations sense the medium when they have a frame to send and send the frame as soon as the medium becomes idle. They monitor the channel for collisions as they send. If there is a collision, they abort the transmission with a short jam signal and retransmit after a random interval. Let us now see how the random interval is determined when a collision occurs, as it is a new method. After a collision, time is divided into discrete slots whose length is equal to the worst-case roundtrip propagation time on the ether ($2\tau$). To accommodate the longest path allowed by Ethernet, the slot time has been set to 512 bit times, or 51.2 μsec.

After the first collision, each station waits either 0 or 1 slot times at random before trying again. If two stations collide and each one picks the same random number, they will collide again. After the second collision, each one picks 0, 1, 2, or 3 at random and waits that number of slot times. If a third collision occurs (the probability of this happening is 0.25), the next time the number of slots to wait is chosen at random from the interval 0 to 23 − 1.

In general, after *i* collisions, a random number between 0 and $2i - 1$ is chosen, and that number of slots is skipped. However, after 10 collisions have been reached, the randomization interval is frozen at a maximum of 1023 slots. After 16 collisions, the controller throws in the towel and reports failure back to the computer. Further recovery is up to higher layers. This algorithm, called **binary exponential backoff**, was chosen to dynamically adapt to the number of stations trying to send. If the randomization interval for all collisions were 1023, the chance of two stations colliding for a second time would be negligible, but the average wait after a collision would be hundreds of slot times, introducing significant delay. On the other hand, if each station always delayed for either 0 or 1 slot, then if 100 stations ever tried to send at once they would collide over and over until 99 of them picked 1 and the remaining station picked 0. This might take years. By having the randomization interval grow exponentially as more and more consecutive collisions occur, the algorithm ensures a low delay when only a few stations collide but also ensures that the collisions are resolved in a reasonable interval when many stations collide. Truncating the backoff at 1023 keeps the bound from growing too large. If there is no collision, the sender assumes that the frame was probably successfully delivered. That is, neither CSMA/CD nor Ethernet provides acknowledgements. This choice is appropriate for wired and optical fiber channels that have low error rates. Any errors that do occur must then be detected by the CRC and recovered by higher layers. For wireless channels that have more errors, we will see that acknowledgements are used. A flowchart representing the binary exponential back off algorithm is given in Fig. 2

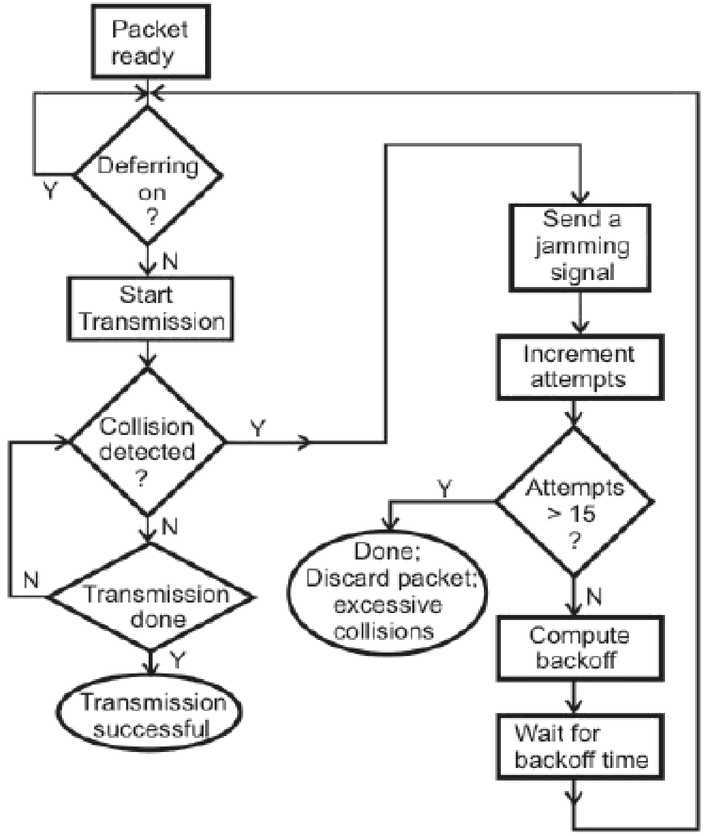


Figure 2: Binary exponential back off algorithm used in CSMA/CD

## 3. MODIFIED BEB ALGORITHM AND TOOLS USED

In this section, we will present tools used in our simulation and the implementation of the changed BEB algorithm.

*NS2 Simulator*

Our simulation is done on NS2. Network simulator 2 or NS2 is an object-oriented discrete event allows us to study the design and protocols for computer networks. It offers various facilities for simulation of the protocols based on TCP, UDP, routing and multi-distribution (broadcast and multicast) in the (wired or wireless) networks. This simulator is a free open source. NS2 is developed in C++ which is a part of the VINT project (Virtual InterNetwork Testbed) is a joint effort led by the University of Berkeley, USC / ISI (University of Southern California's Information Sciences Institute), the LBL (Lawrence Berkeley National Laboratory) and Xerox PARC. It is supported by DARPA (Defense Advanced Research Projects Agency). It uses IU OTCL interpret. Through this language, the user can describe conditions of the simulation: network topology, selected from a lot of physical links, used protocols, communications done, etc.. The user can also create new objects in C++ and use them in NS by instantiations with OTCL, the two languages C++ and OTCL have both very close hierarchies to each other. In NS2 several libraries are available and there are some specific needs for simulation of wireless network and "multicast" communication [Opn07][Ane99][Ns07][Ns06][Luc03]. In our work we use NS2 for its flexibility and the availability of its code [Ben07].

*Communication entity in NS2*

The node (communicating entity) is the basic element of our model. A node in NS2 is a class defined in OTCL which has three entities containing: the classifier, the link and the agent.

*The classifier*

The function of a node of the fields is examined by received packet, and more specifically, the source address and destination address. According to contention losses, the node sends the packet on its outgoing interfaces (Fig. 3). In NS2, then this is performed by an object called "Classifier". There are several types of classification that are used for different purposes:

- "Address classifiers" is used to treat with unicast packets, and its role is directly to select packets addresses, direct the node, and select the link to the next node.
- "Port classifier" its role is to select the agent to the packet which is intended.
- "Multicast classifier" is used to classify multicast packets.

*The link*

It is used to connect the nodes by each other (Fig. 3). A link is defined by several parameters including: bandwidth, entry point, the lifetime of each packet, etc.. NS2 has several types of links, so we can distinguish unidirectional links from bidirectional links and wired links to wireless networks model without sound.

*The Agent*

Agents represent endpoints where packets in network layer are constructed and consumed. These agents are the third component of the node. In NS2, the agent's role is to provide the destination address; its function is to generate the packets and the interface to the application class (Fig. 3).
In NS2 there are several types of agents, each has a specifying role:

- TCP agent: for emitting TCP traffic
- UDP agent: for emitting UDP traffic
- TCPSink agent: for the receipt of TCP traffic
- NULL Agent for receiving UDP packets.

Figure 3 shows the Existing entities in a node and the links between ITS entities.

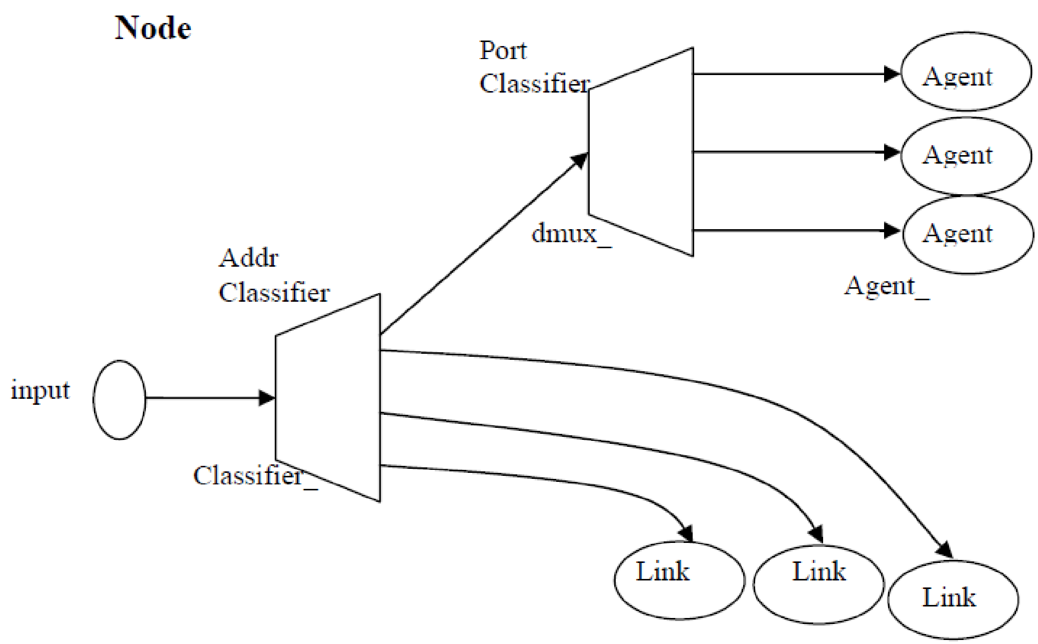


Figure 3: Diagram of a node in NS2.

## 4. TCL LANGUAGE

Tcl (originally from "Tool Command Language") is two things: a scripting language, and an interpreter for that language that is designed to be easy to embed into our application. Tcl and its associated graphical user-interface toolkit, Tk, were designed and crafted by Professor John Ousterhout of the University of California, Berkeley. The Tcl interpreter has been ported from UNIX to DOS, Windows, OS/2, NT, and Macintosh environments. The Tk toolkit has been ported from the X window system to Windows and Macintosh.

## 5. AWK LANGUAGE

**Awk** is a programming language that handles several tasks, like mechanical data manipulation - changing the format of data, checking its validity, finding items with some property, adding up numbers, printing reports, and the like, with very short programs, often only one or two lines long. An awk program is a sequence of patterns and actions that tell what to look for in the input data and what to do when it's found. Awk examines a set of files for lines matched by any of the patterns; when a corresponding line is found, the corresponding action is performed. A pattern can select lines by combinations of regular expressions and comparison operations on strings, numbers, fields, variables, and array elements.

Actions may perform arbitrary processing on selected lines; the action language looks like C but there are no declarations, and strings and numbers are built-in data types.

## 6. SIMULATIONS

Configuration of communicating entities
*Configuration of access point*

The base station is configured as follows (Fig. 4)

$ Ns_ node-config-adhocRouting DSDV

- llType-LL / / LL layertype
- macType Mac/802_11 \ / / type of the MAC layer
- IfqLen 800 \ / / length of the tail
- antType Antenna / OmniAntenna \ / / type of antenna
- phyType Phy / WirelessPhy \ / / type of physical layer
- channelType Channel / WirelessChannel \ / / type of channel
- wiredRouting ON \ / / Wired link with other AP's.

In this list we find the basic configurations of the AP. (Type of MAC and physical layer, antenna type...). Figure 4 describes entities used for the AP (agents, links, and classifiers).

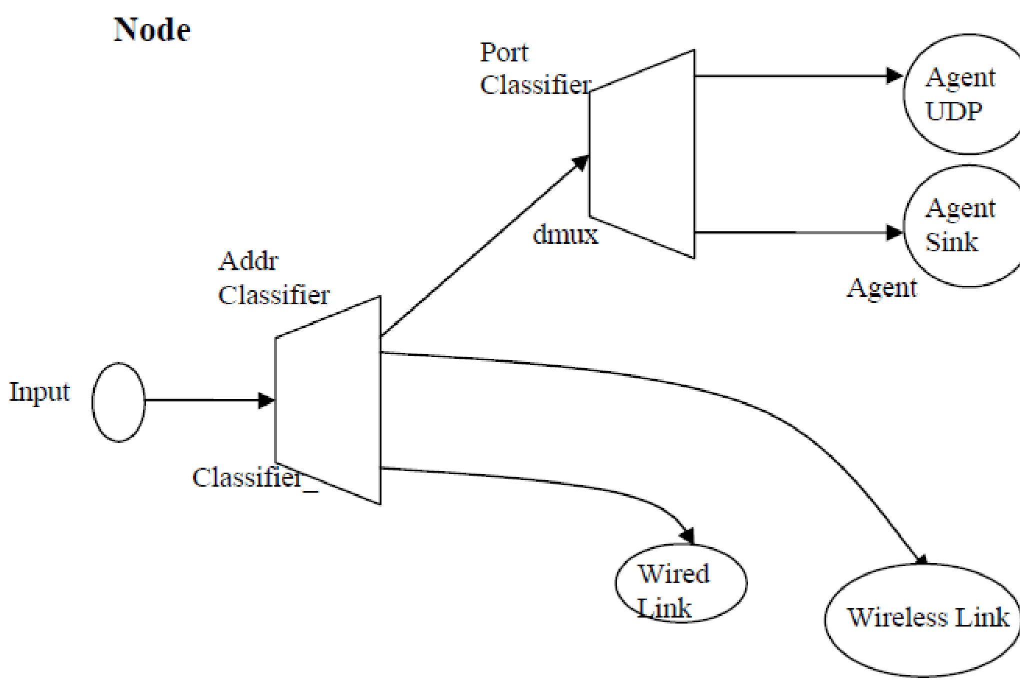


Figure 4: Figure of base station

*Configuration of mobile nodes*

The nodes are moving in the $M_i$ cell base station where each node has a unicast address and exchange packets with other mobile station through the base station. In each node there exist a single wireless link but there are two agents (Fig. 5):

- An agent to send UDP packets (UDP Because in Our case, we use UDP only)
- A null Agent to Receive UDP packets.

These nodes operates in the cell of an access point to which they are attached as shown in Fig 5.

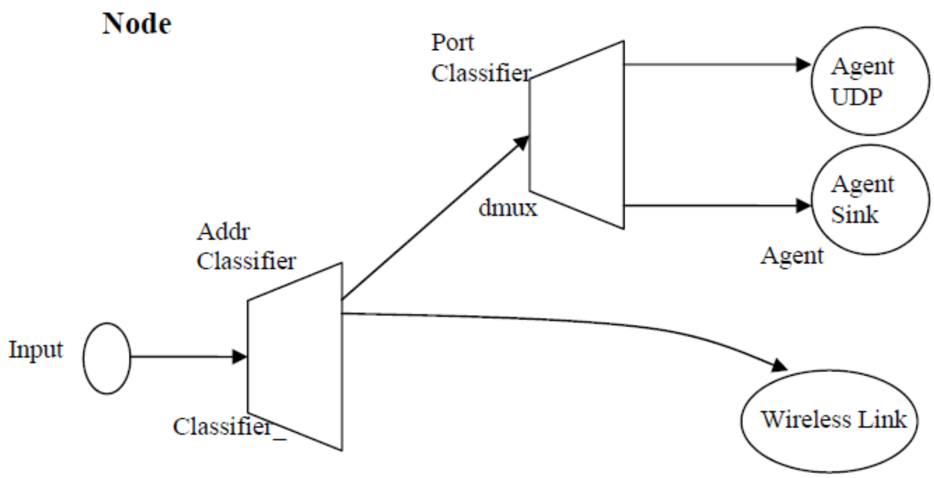


Figure 5: Diagram of the mobile node

*General parameters*

The general parameters of this simulation are given in (table 1). These parameters for the simulation model are compatible with the measure made.

Table 1: Simulation Parameters

| | |
|---|---|
| CBR Packet Size | 210 bytes |
| CWMin | 31 ms |
| CWMax | 1023 ms |
| SIFS | 0.000010 s |
| SlotTime | 0.000020 s |
| CBR rate | 448Kb |

## 7. MODELING THE METHODS OF BACK-OFF

We will now describe the numerical experiments to adjust the simulation parameters “a” and “b” of the back-off algorithm.

The purpose of this simulation is to study the effect of the load (by changing the number of mobiles) on the transmission time for different values of parameters: “a” and “b”. Our approach is based on cooperation involving the mobiles in the cell, by pairs, and each pair in the cell transmits CBR traffic to other one during a specified time.

Our simulation protocol is the following: We will evaluate the changes in throughput and end to end delay while changing “a” and “b” and the number of nodes in the cell. To increase the cell load, we increase the number of mobiles in introducing new pairs (the mobile communicates with other in pairs), each forming a cooperation group. In the example in (Figure 6) it is pair M5 - M6. For each simulation we vary the values of “a” and “b”. All these mobiles broadcast via the AP (in pairs) according to NS2 simulation parameters mentioned above.

If D is the transmission delay between mobile references, it can be defined as time between sending the message (Temis) and corresponding time to receive the message by the mobile receiver Trec: **D= Max(Trec - Temis )**

It should be noted that this simulation will help us to find the optimal values of “a” and “b” by changing the Back-off algorithm.The figure below describes our work

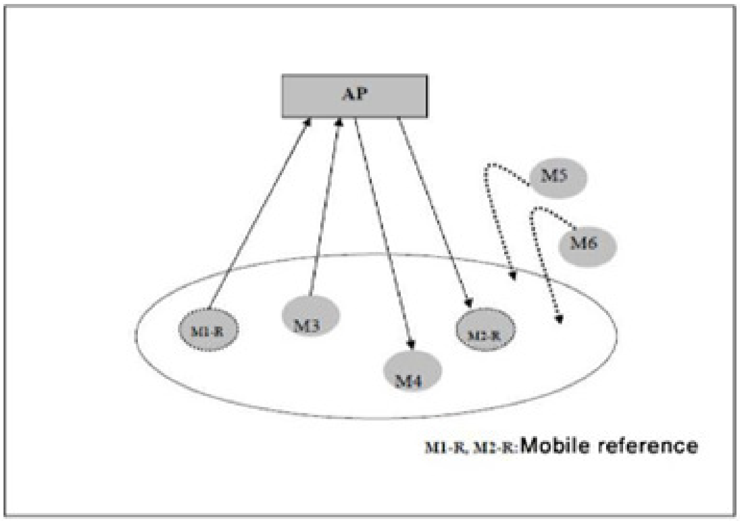


Figure 6: Principle of simulation used to assess the changed Back-off

## 8. IMPLEMENTATION OF THE MODIFIED BEB

In fact the back-off algorithm BEB is implemented in standard NS2. To implement our "modified" algorithm of Back-off algorithm it is sufficient to act on some files in the hierarchy of NS2 MAC layer. We show this hierarchy in the (Figure 7).

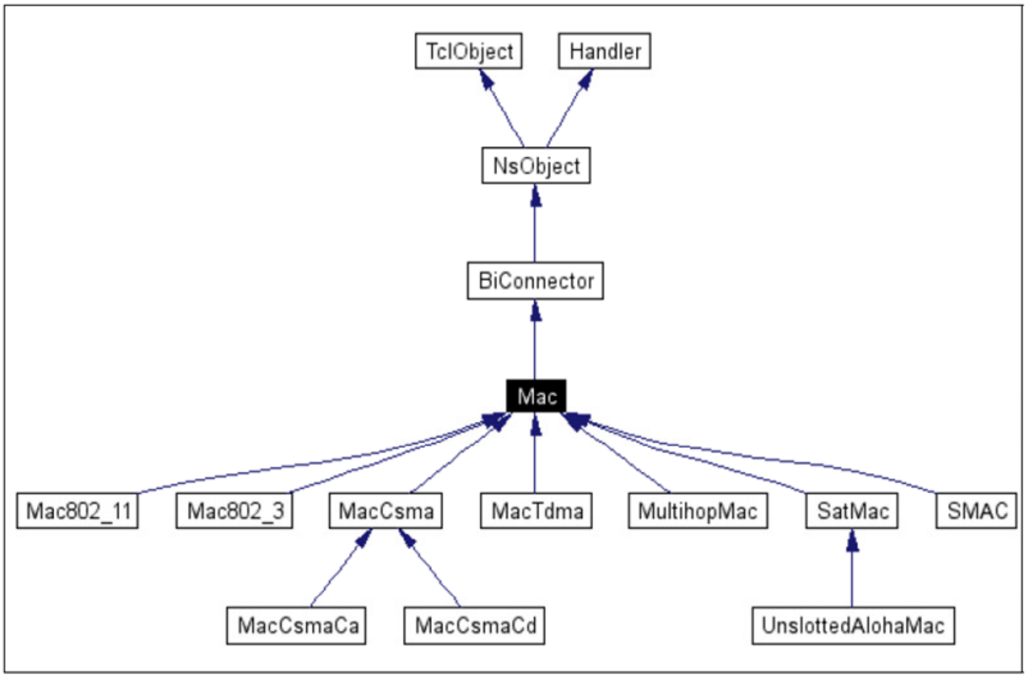


Figure 7: Hierarchy of NS2 MAC layer [Ns06].

After the study of this hierarchy, we decided to modify the code in C++ functions included in in the following files:

- tcl/lan/ns-mac.tcl.
- tcl/lib/ns-lib.tcl.
- Mac/mac-802_11.h.
- Mac/mac-802_11.cc.
- tcl/lib/ns-default.tcl.

We want to change the formula of Back-off as follows:

On failure of transmission: CW = min (a.CW, CWmax)
and each successful transmission, we decrement the value of CW by value b.
CW = max (CW-b, CWmin)
To make these changes we use the C++ of NS2 and we change the File Mac/Mac-802_11.hh as follows:
In case of transmission failure, we modified the inc_cw() function by setting CW=min(a* CW,CWmax), a is initialized with a value taken from the interval [1,3].

*Original Function:*

```
inline void inc_cw()
    {
                cw_ = (cw_ << 1) + 1;
                if(cw_ > phymib_.getCWMax())
                        cw_ = phymib_.getCWMax();
          }
```

*Modified one:*

```
inline void inc_cw()
        {
        cw_= min((u_int32_t)(phymib_.geta()*cw_),phymib_.getCWMax());
        }
```

In case of successful transmission, we modified the rst_cw() method by setting CW=max(CWmin, CW-b) instead of CW = CWmin.

*Original Function:*

```
inline void rst_cw()
        {
                cw_ = phymib_.getCWMin();
        }
```

*Modified Function:*

```
inline void rst_cw()
        {
cw_= max(phymib_.getCWMin(),(u_int32_t)(cw_- phymib_.getb()));
        }
```

## 9. SCENARIO OF SIMULATION

We calculate the average transmission time between all mobiles during a time interval T = 120s for which the cell load remains constant.

To find the best values of “a” and “b”, we vary b in the interval [0, 10] steping 1 each time and for each b, value varies in interval [1, 3] steping 0.1 each time. The results obtained at the end of the simulations corresponding to different values of “a” and “b” will allow us to choose their optimal values. The optimum means here leading the minimum transmission delay, and maximum throughput.

The following flowchart (Figure 8) describes the scenario simulation and the increment of “a” and “b”.

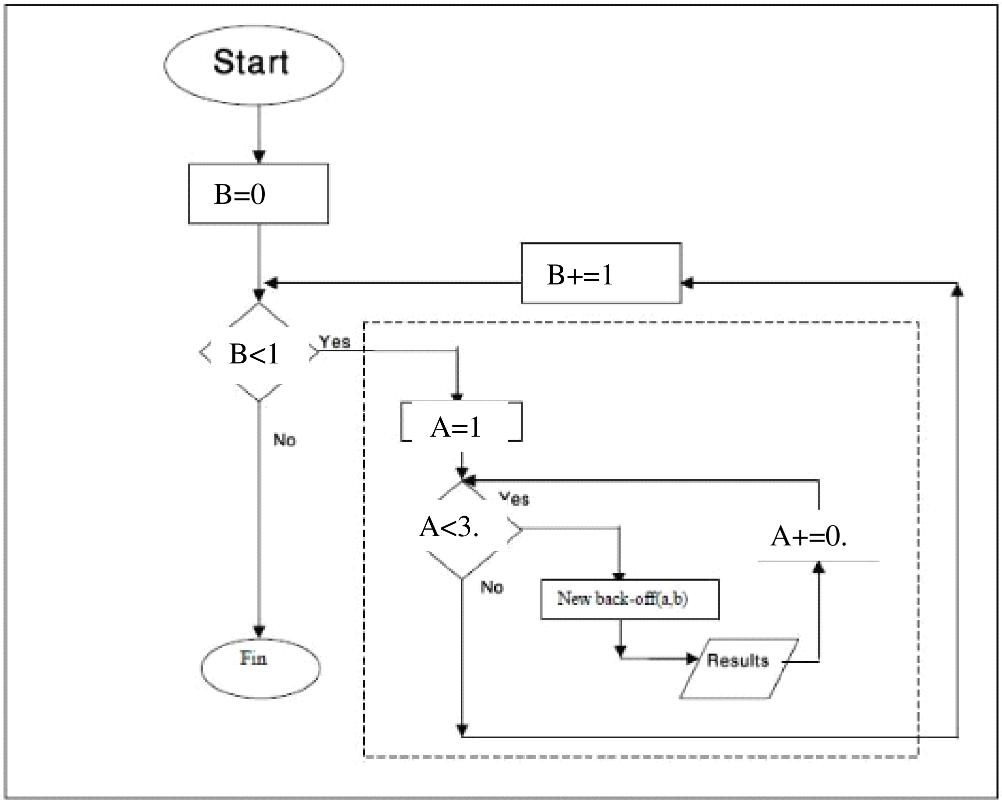


Figure 8: Algorithm of evaluating (a, b) values of Back-off changed.

# 9. ANALYSIS OF RESULTS AND VALIDATION

In this section, we present the results obtained by changing the mechanism of Back-off. First we present test result for base BEB and the modified BEB with the same scenario mentioned in chapter 3 but during a period of 2s and not 120s for testing purposes. Second we present the results of our end simulation and obtaining the optimum values of "a" and "b".

Before applying the new algorithm we make a simulation test to get results in order to proof that the new added parameters in ns2 files takes effect. The test is applied with the parameters mentioned in the previous chapter but during a period of 2 s only. The BEB algorithm without change has a=2 since when a failer transmission happens the contention window is doubled that's means multiplied by 2 leading to a=2 but when a successful transmission occur cw returns to cwmin and in our algorithm cw will be the maximum between cwmin and cw-b for this reason we cannot deduce b but we will do the simulation assuming b is 0 and table 5.1 shows the results which proves the effect of the new algorithm on throughput and delay.

Table 2: Comparing results between old and new algorithm

| Throughput Modified Algorithm | | | | | |
|---|---|---|---|---|---|
| | a | b | T. RecvByte | Throuput | r/s |
| n=6 | 2 | 0 | 6.64 | 435.63 | 0.19 |
| n=10 | 2 | 0 | 6.79 | 445.97 | 0.11 |
| n=30 | 2 | 0 | 4.64 | 304.84 | 0.02 |
| Delay Modified Algorithm | | | | | |
| | a | b | Packet Delivery Ratio | Total Dropped Packets | Average End-to-End Delay |
| n=6 | 2 | 0 | 19.3 | 177 | 308.33 ms |
| n=10 | 2 | 0 | 10.98 | 453 | 317.53 ms |
| n=30 | 2 | 0 | 2.33 | 2531 | 335.18 ms |
| Throughput UNModified Algorithm | | | | | |
| | a | b | T. RecvByte | Throuput | r/s |
| n=6 | 2 | 0 | 7 | 459.83 | 0.2 |
| n=10 | 2 | 0 | 6.9 | 452.06 | 0.11 |
| n=30 | 2 | 0 | 4.61 | 302.61 | 0.02 |
| Delay UNModified Algorithm | | | | | |
| | a | b | Packet Delivery Ratio | Total Dropped Packets | Average End-to-End Delay |
| n=6 | 2 | 0 | 20.34 | 174 | 355.53 ms |
| n=10 | 2 | 0 | 11.14 | 544 | 388.19 ms |
| n=30 | 2 | 0 | 2.32 | 2573 | 417.30 ms |

## 10. NEW BEB ANALYSIS AND RESULT

Our goal is to make several simulations by changing the values of “a” and “b” in modified Back-off mechanism. In each simulation, we measure the transmission time between each two mobiles and calculate the average end to end delay of communicating mobiles, besides we calculate the throughput by measuring the total transmitted bits during a specified period of time (simulation period).

Our goal is to choose the optimal values of “a” and “b”, which gives:

- The minimum transmission time
- Maximum capacity.

For this, we made 21 x 11 simulations for the values of “a” and “b”. We present in the appendix A the tables of the results obtained for throughput and for end to end delay for which we can choose the optimal values of “a” and “b”. Figurs 9, 10, 11, 12, 13, 14 shows these results graphicaly.

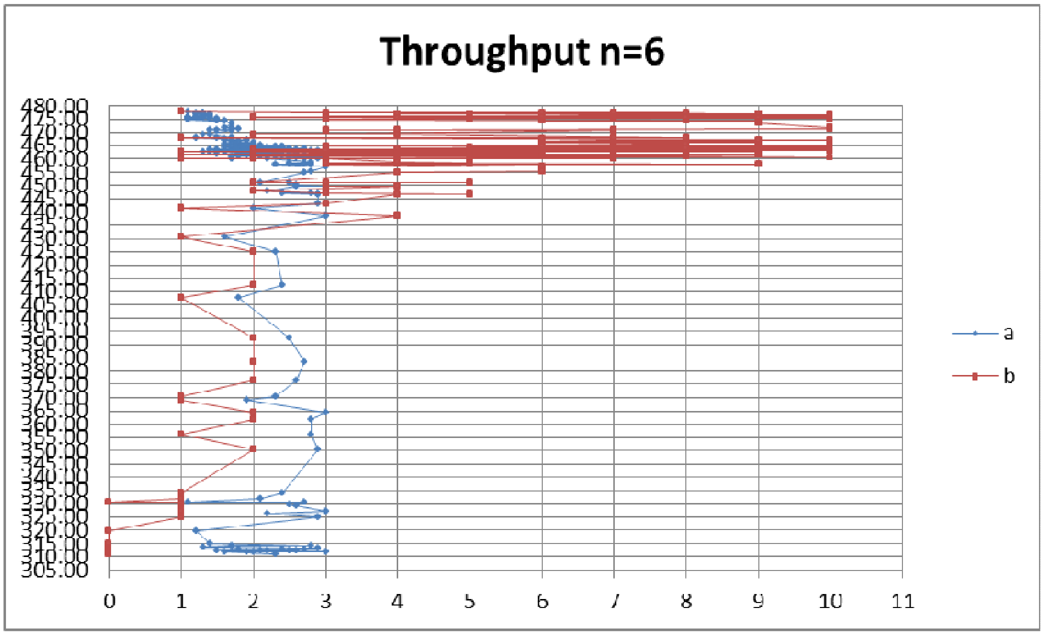


Figure 9: Throughput for no. of nodes =6

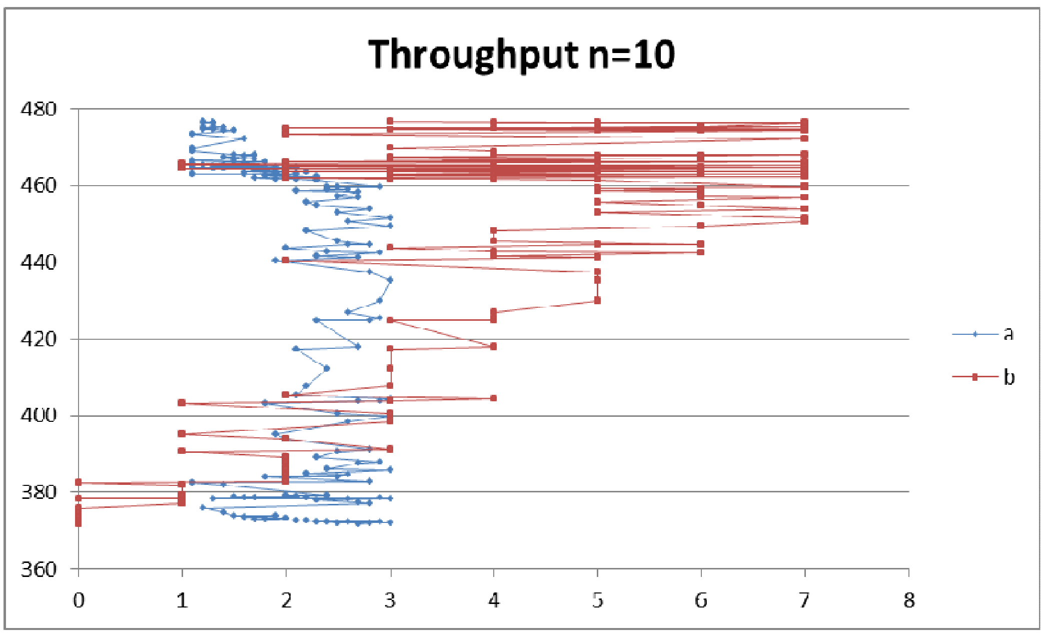


Figure 10: Throughput for no. of nodes =10

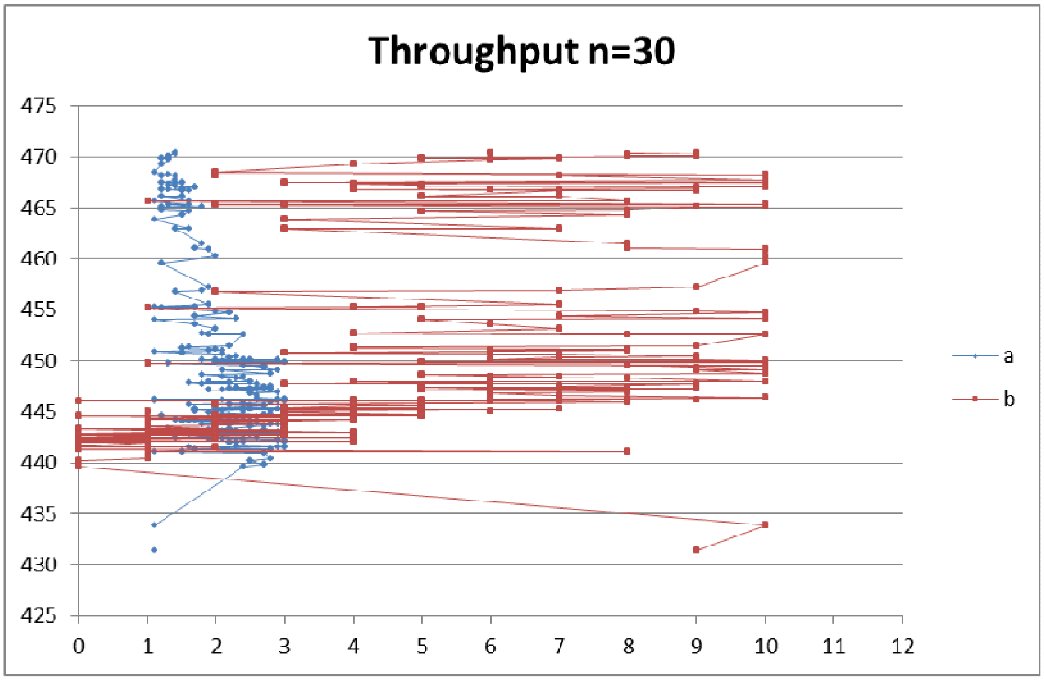


Figure 11: Throughput for no. of nodes =30

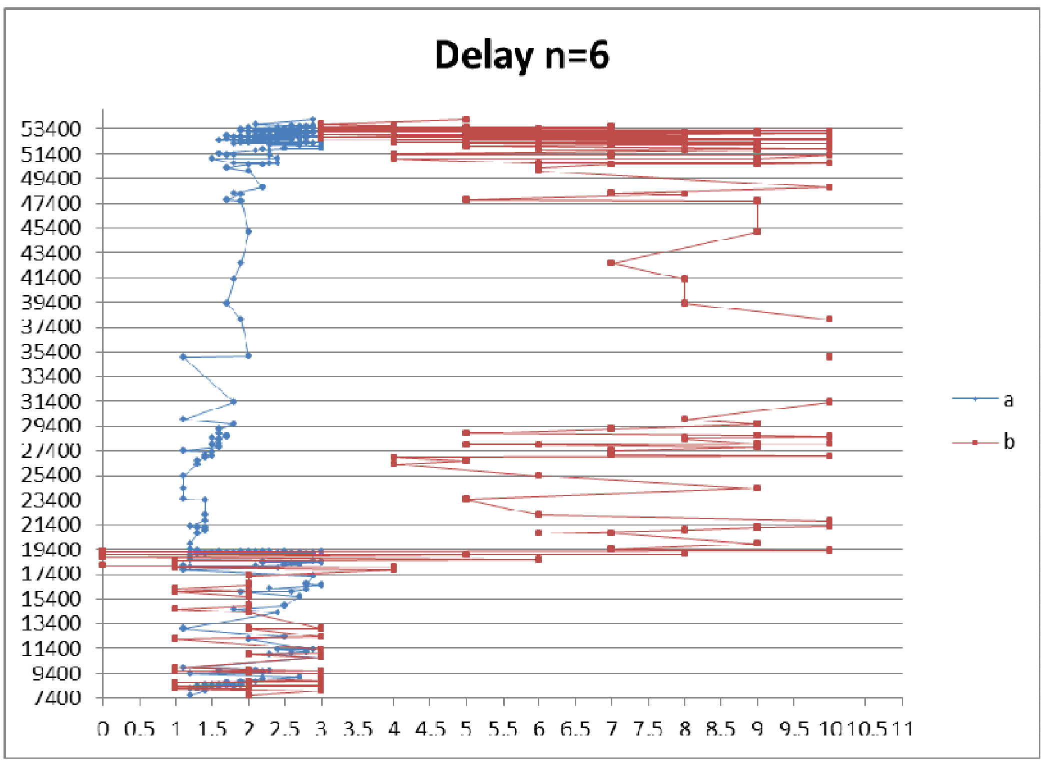


Figure 12: End to end Delay for no. of nodes =6

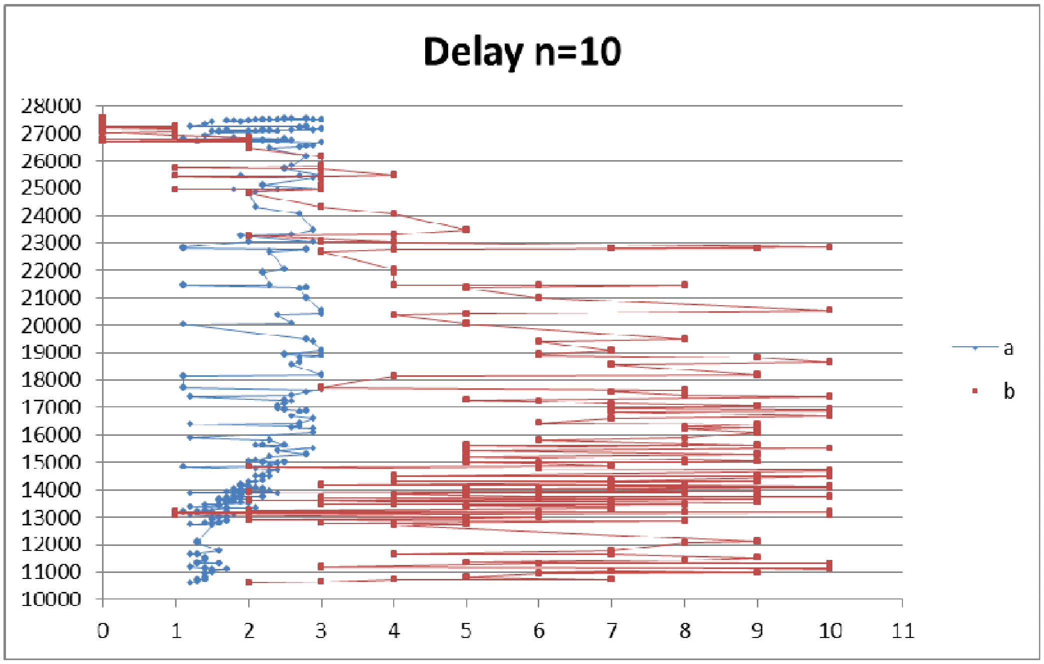


Figure 13: End to end Delay for no. of nodes =10

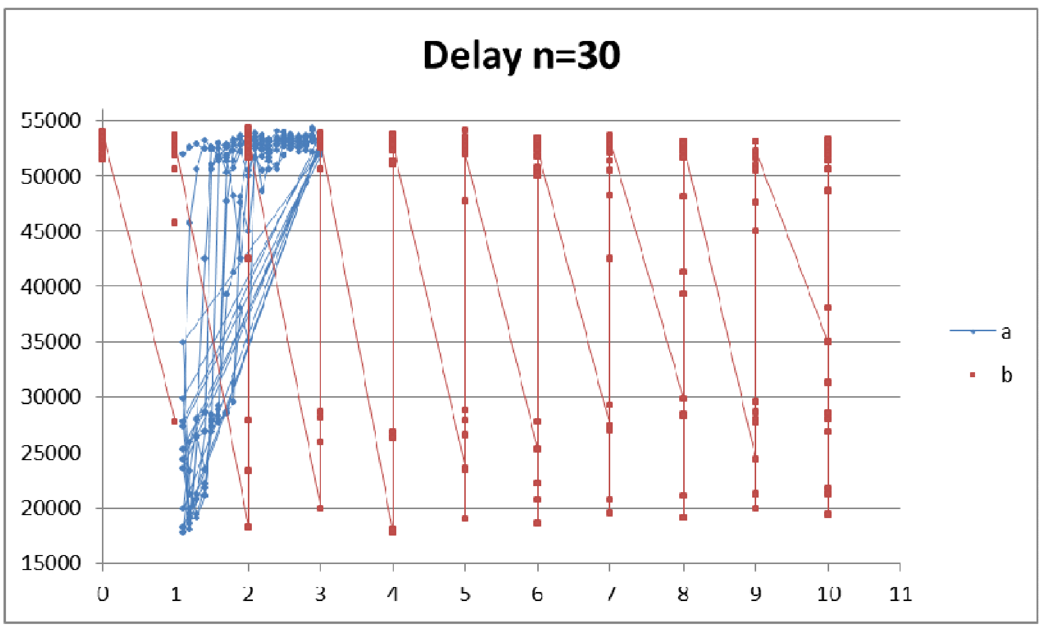


Figure 14: End to end Delay for no. of nodes =30

According to the graphs shown and the tables in Appendix A we deduce that the optimum values of "a" and "b" are:

According to throughput measurement:

1 For n=6: a=1.1, b=1
2 For n=10: a=1.2, b=3

3 For n=30: a=1.4, b=6

According to delay measurement:

1 For n=6: a=1.2, b=2

2 For n=10: a=1.2, b=2

3 For n=30: a=1.1, b=4

The simuation result lead us to more than one optimum values of “a” and “b” according to circumstances the network based on.

## 11. GENERAL CONCLUSION

This paper aims to minimize the delay time through wireless communication by changing the BEB algorithm. The Protocol CSMA / CA is a protocol that manages access to the radio channel by performing an arbitration based on time. This causes many problems in relation to time transmission between mobiles moving in a cell 802.11. what we have done show that the protocol using CSMA / CA access time believed rapidly when the number of stations and / or the network load increases or other circumstances affects the network.

Our objective is to minimize the time transmission cycle of the information between mobiles moving in a Wi-Fi. We reach our solution by changing two main functions in the BEB algorithm and our study proves that the changes we made give an acceptable result. We made the simulation according to variable number of nodes and for each one we get a new values of “a” and “b”.
Future studies will be to create a fuzzy logic function to make choice of the optimum values of “a” and “b” according to specific rules and parameters.

## REFERENCES


[1] [Ahm99]H. Ahmadi, A. Krishna, and R.O. LaMaire, « Design issues in wireless LANs”, Journal of High Speed Networks, Personal Wireless Communication, 1999.

[2] [Ane99]P. Anelli, E. Horlait, « Ns-2: principle », version 1.3, 1999.

[3] [Ben07]Sabri Benfarhat, « rapport interne équipe réseau et protocole », LIMOS, Clermont Ferrand, 2007.

[4] [Bia00]Giuseppe Bianchi, “Performance Anaylsis of the IEEE 802.11 Distributed Coordination Function”, IEEE Journal on Selected Areas in Communications, Vol. 18, Number 3, March 2000.

[5] [Bia96]Giuseppe Bianchi, Luigi Fratta, Matteo Oliveri, “Performance evaluation and enhancement of the CSMA/CA MAC protocol for 802.11 wireless LANs”, in: Proc. of IEEE PIMRC'96, October 1996.

[6] [Bin02]Benny Bing, “Wireless Local Area Networks”, Wiley-Interscience, 2002.

[7] [Bou04]Pierre-Olivier BOURGEOIS - Alexis MARCOU, « Security of IEEE 802.11 », août 2004.

[8] [Byu05]Byung-Jae Kwak, Nah-Oak Song, Miller, “Performance analysis of exponential backoff”, Networking, IEEE/ACM Transactions on , volume 13, issue 2, april 2005.

[9] [Car03]M. M. Carvalho, J. J. Garcia-Luna-Aceves, “Delay analysis of IEEE 802.11 in single-hop networks”, Proc. of IEEE ICNP, Nov. 2003.

[10] [Car06]Carlos Cardeira, Ronald Schoop, « Automation of wireless networks », Jautomatise N° 48 - Septembre-Octobre 2006.

[11] [Cha04]Chae Y. Lee, Hee K. Cho, “Multiple multicast tree allocation in IP network”, Computers & Operations Research, Volume 31, Issue 7, June 2004.

[12] [Che98]K. Cheun and S. Kim, “Joint delay-power capture in spread-spectrum packet radio networks”. IEEE Transaction on Communications, 1998.

[13] [Chi06]J. Chittamuru, A. Ramanathan, M. Sinha, “Simulation of Point Coordination Function for IEEE 802.11 Wireless LAN using Glomosim”, 2006.

[14] [Chu05]Chunyu Hu, Hwangnam Kim, and Jennifer C. Hou, “An Analysis of the Binary Exponential Backoff Algorithm in Distributed MAC Protocols”,UIUCDCS-R-2005-2599 (UILU-ENG-2005-1794), July 2005.

[15] [Cirr02]"Wireless Communications for Industrial Applications", Cirronet White Paper, Summer 2002.
[16] [Dev93]Daniel DEVARSIRVATHAM, Theodore RAPPAPORT, "Radiowave propagation measurements & modeling for personal communications", Tutorial 4, IEEE International Conference on Communications ICC'93, Geneva, Switzerland, May 27, 1993.
[17] [Ebe00]Jean-pierre Ebert, Adam Wolisz, "A performance comparison of point and distributed coordination function", MOMUC, 2000.
[18] [Erg04]Sinem Coleri Ergen, "ZigBee/IEEE 802.15.4 Summary", Berkeley, September 2004
[19] [Fah05]FAHS Walid, BAKHACHE Bacem , JACQUET Frédérique, MISSON Michel, « Stratégies de Multicast contraint par le temps au sein d'un WLAN », JDIR 2005, proceeding P.108-115.
[20] [Fah06a]Walid Fahs, Bassem Bakhache, Frédérique Jacquet, Michel MISSON, "Multicast Strategies for Robots moving within the coverage of a WLAN", VTC Fall 2006, Canada.
[21] [Fah06b]W. Fahs, B.Bakhache, M. Misson, and F. Jacquet, "Study of a Wireless Medium Access Method inspired from PCF for industrial applications", IEEE ICTTA Spring 2006,Ref. ICTTA405.
[22] [Far06]Jahanzeb Farooq, Bilal Rauf, "Implementation and Evaluation of IEEE 802.11e Wireless LAN .in GloMoSim", rapport de master, 2006
[23] [Fre95]Antonio FREITAS, « Etude de la transposition de la méthode d'accès « jeton sur bus» pour un réseau de stations nomades dialoguant par infrarouge dans une configuration multisauts », Thèse de Doctorat, Université Blaise Pascal, 1995.
[24] [Fri46]Friis, H.T., "A Note on a Simple Transmission Formula", 1946.
[25] [Ger91]E. Geraniotis and M. Soroushnejad, "Probability of Capture and Rejection of Primary Multiple Access Interference in Spread Spectrum Networks", IEEE Trans.on Communications, 39(6), 1991.
[26] [Gou67]G. Goudet, « applications des équations de Maxwell », éditions Masson 1967.
[27] [Goo88]Jonathan Goodman, Albert G. Greenberg, Neal Madras, Peter March, "Stability of binary exponential backoff", Journal of the ACM (JACM), Volume 35 , Issue 3 (July 1988) Pages: 579 - 602 .
[28] [Har04]Lawrence J. Harte, "Introduction to Bluetooth, Technology, Market, Operation, Profiles, and Services", Althos Publeshing, 2004.
[29] [Her04]Sébastien Hernandez, Patrick Lafargette, Antonio Freitas, Michel Misson, "FIRST evaluations of a simulation Architecture of the Use of an IEEE 802.11 WLAN in industrial Context", IEEE Wireless Communications and networking conference, Atlanta 2004.
[30] [Her06]Sebastien Hernandez, Evaluation et optimisation du mécanisme de 'Handover' dans un réseau local sans fil dédié aux applications à trafic contraint par le temps, Thèse de Doctorat, Université Blaise Pascal, 2006.
[31] [Hpl07]http://www.hpl.hp.com/personal/Jean_Tourrilhes.
[32] [Iee97]IEEE 802.11-1997, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY)", 1997.
[33] [Itu05]Recommendation ITU-R P.1238-4, "Propagation data and prediction method for the planning of indoor radio communication systems and local area networks in the frequency range of 900 MHz to 100 GHz", 2005.
[34] [Ixa06]http://www.ixarm.com/Fiche-industrie-de-CYBERNETIX.
[35] [Jac04]F.Jacquet, P.Lafargette, P. Llamas, M.Misson, "A multicast Approach for exchange between co-operating mobiles in an 802.11 WLAN", The 8th Multi-Conference on systemic cybernetics and informatics SCI 2004, Proceeding, pp 7-12, Orlando, USA 2004.
[36] [Lep91]F.LEPAGE et al., « Local industrial applications », 2nd edition, HERMES,1991.
[37] [Liu00]Jane W. S. Liu, "Real-Time Systems, University of Ilinois at Urbana-hampaign", Person education, 2000.
[38] [Luc03]Gilberto flores Lucio, Marcos Paredes-Farrera, Emmanuel Jammeh, Martin Fleury,Martin J.Reed, Opnet Modeler and NS2: comparing the accuracy of network simulator for packet-level analysis using a network testbed, WSEAS Transactions on Computers, Issue 3,Volume 2:700-707 July 2003.
[39] [Man05]Saher S. Manaseer, Mohamed Ould-Khaoua, "Logarithmic Based Backoff Algorithm for MAC Protocol in MANETs", 2005.
[40] [Mar02]P. S. Marshall; "Industrial Ethernet", ISA Press, (ISBN 1-55617-869-7), 2002.
[41] [Mcc06]Patrick McCurdy, Ira Sharp, "Exploring New Wireless Technologies for Industrial Applications", Presented at ISA Expo 2006.

[42] [Met76]J.J. Metzner, “On improving utilization in ALOHA networks”. IEEE Trans. Commun., page 447,1976.
[43] [Mis05]Michel Misson, cours de Master Réseau, 2005.
[44] [Muh02]Paul Mühlethaler, « 802.11 et les réseaux sans fil », Eyrolles, 2002.
[45] [Mut89]K. Mutsuura, H. Okada, K. Ohtsuki, and Y. Tezuka, “A New control scheme with capture effect for random access packet communications”, In Proc. IEEE GLOBECOM, page 938, 1989.
[46] [Nak05]Nakjung Choi, Yongho Seok, Yanghee Choi, Sungmann Kim, Hanwook Jung, “Enhanced backoff scheme for the IEEE 802.11 DCF”, Vehicular Technology Conference,2005. VTC 2005-Spring.Volume 3, Issue , 30 May-1 June 2005 Page(s): 2067 - 2070 Vol. 3.
[47] [Ns06 ]Tutorial for the Network Simulator NS, http://www.isi.edu/nsnam/ns/tutorial/,2006.
[48] [Ns07] http://www.isi.edu/nsnam/ns/.
[49] [Oba98]Shuichi Obayashi, “A Body-Shadowing Model for Indoor Radio Communication Environments”, 1998.
[50] [Opn07]www.opnet.com.
[51] [Pil04]Jean-François Pillou, « introduction to Wifi (802.11) », 2004.
[52] [Pro07]John G. Proakis, Massoud Salehi, “Digital communications”, McGraw Hill Higher Education, 2007
[53] [Rap96]T. S. Rappaport, “Wireless communications, principles and practice”, Prentice Hall,1996.
[54] [Sta01]William Stalling, “Local Networks”, Sixth Edition, Macmillan, 2001.
[55] [Std99]SO/IEC Std 8802-11, “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specification”, 1999.
[56] [Tia03]Tian, S. Csaba, T. Lohmar, H.-Y. Lach, R. Tafazolli, C. Janneteau, “comparison of three approaches towards mobile multicast”, Y. IST Mobile Summit 2003, Aveiro, Portugal,16-18 June 2003.
[57] [Vas04]A. Vasan, A. Kochut, and A. U. Shankar, “Sniffing out the correct Physical Layer Capture model in 802.11b”, Technical Report UMIACS-TR-2004-26 and CS-TR-4583,Department of Computer Science, University of Maryland College Park, April 2004.
[58] [Vel03]Hadzi-Velkov, Z., Spasenovski, B., “An analysis of CSMA/CA protocol with capture in wireless LANs”, Wireless Communications, March 2003.
[59] [War01]C. Ware, J.F. Chicharo, and T. Wysocki, “Modelling of capture behaviour in ieee 802.11 radio modems”, IEEE International Conference on Telecommunications, 2001.