

Optimized Pessimistic Fibonacci Back-off Algorithm (PFB)

Muneer Bani Yassein
College of Computer and
Information Technology
JUST Amman, Jordan

Mohammed Ahmed Alomari
College of Computer and
Information Technology
JUST Amman, Jordan

Constandinos X. Mavromoustakis
Department of Computer Science
University of Nicosia
Nicosia, Cyprus

Abstract— MANET is a self-directed system consisting of mobile nodes, which can be either routers and/or hosts. Nodes in MANET are connected by wireless links without base stations. The Backoff algorithm considered as a main element of Media Access Control (MAC) protocol, which is used to avoid collision in MANET's. The Fibonacci Backoff algorithm and the Pessimistic Fibonacci Backoff are proposed to improve network performance depending on contention window size. This research introduces a new hybrid Backoff algorithm called Pessimistic Fibonacci Backoff (PFB) Algorithm which merges the two previous algorithms in order to find the most proper contention window sizes that reduce collisions as much as possible. This research takes into consideration and evaluates each of the following main measurements: Packet delivery ratio, normalized routing load and end-to-end delay. Based on the extracted simulation results, PFB algorithm outperforms Pessimistic Linear-Exponential Backoff (PLEB) by up to 76%,40.41%, 31.88% in terms of Packet delivery ratio, end-to-end delay and normalized routing load respectively, especially in the sparse environments. All of the simulation results are obtained by the well-known NS-2 Simulator, version 2.34, without any distance or location measurements devices.

Keywords- Back-off; collision; end-to-end delay; normalized routing load; packet delivery Ratio; MANET's; PLEB, PFB; and MAC.

I. INTRODUCTION

Wireless networks (WN) use radio signals to communicate among computers and other network devices. WN's are getting popular nowadays due to easy setup feature. World is now moving to this type of communication. Today, this vision is being challenged by various forms of mobility, which are effectively reshaping the landscape of modern distributed computing. Mobile wireless networks consist of two kinds of mobile networks, infrastructure-based and ad-hoc wireless networks.

Wireless sensor networks (WSN) corresponds as a communication way and supports the random movement of the nodes. The main features and challenges of WSN are [8]: low cost devices, large scale of deployment, end-to-end quality of service, energy-efficient devices, and secure operation.

A Mobile ad-hoc Network (MANET) is a self-directed system consisting of mobile nodes (including routers and hosts) connected by wireless links. MANETs have received substantial attention due to their strong features, such as

Multi-hop Routing, distributed operations, dynamic topology, capacity and light-weight terminals [1,2].

The Media/Medium Access Control (MAC) protocol is a sub-layer of the data link layer, which provides a control mechanism to allow packet transmission through the wireless network. Within MANETs, MAC protocol consists of several key parts, such as the Backoff mechanism which solves the collision problem which occurs when more than one node transmits data simultaneously due to single transmission restrictions through the channel [3,4]. To realize how the infrastructure wireless networks function, if there are two computers, each one contains a wireless adapter to connect with an access point. When data is transmitted using a wireless router as a binary data, then the operation at the receiver will follow a vice procedure [4, 5]. The base stations are the bridges within these networks, having a role of [1, 6]: linking each mobile node with the nearest base station (Within node Range) and allowing the communication between them.

This research proposes a Backoff algorithm called Pessimistic Fibonacci Backoff (PFB) Algorithm to reduce the differences between contention window sizes using delay parameter, normalization and Packet Delivery Ratio. Its efficiency measures by merging more than one increment behavior in order to have long waiting times when a collision suddenly occurs. This algorithm uses the Pessimistic Linear Exponential Backoff algorithm introduced in [7] where the structure replaces the linear and exponential waiting time with an exponential, cubic, and a Fibonacci series.

The rest of this paper is organized as follows; Section II covers the motivations behind this work, Section III covers the methodologies and presents the related work in order to give a better understanding for the Back off algorithms utilized in the past by other researchers. Section IV presents the proposed Pessimistic Fibonacci Back off algorithm proposed whereas Section V covers the results and introduces the analysis of these results. Finally, the last Section concludes the paper in the last section.

II. MOTIVATIONS

In MANETs, the Backoff field considered to be one of the researcher's main areas in order to achieve an efficient Backoff algorithm [3]. In literature, many Backoff algorithms were proposed such as: Binary Exponential Backoff (BEB), Fibonacci Increment Backoff (FIB), Logarithmic Backoff (LOB), Pessimistic Linear-Exponential Backoff (PLEB) and

Optimistic Linear-Exponential Backoff (OLEB). Binary Exponential Backoff is considered as standard Backoff algorithm in MANETs.

The PLEB Algorithm used in IEEE 802.11 MAC protocol provides exponential and linear increments in contention window values depending on Backoff timer (BOT). These two incremental behaviors make BOT increasing quickly. Therefore, the need to develop a powerful Backoff algorithm is more timely in Wireless Networks in order to enable more features such as the desired stability, the minimum network overhead, the power consumption and the maximum network throughput. All these factors lead to achieve the better network performance.

III. RELATED WORK

The Fibonacci Backoff Algorithm reduces the differences between concurrent contention window sizes using a mathematical Fibonacci series as in equation 1 to reduce the difference between successive contention window sizes.

$$F(n)=Fn-1+Fn-2 \text{ where } F0=0 \text{ and } F1=1 \quad (1)$$

F(n) represents the new contention window size, leading to a smaller increment on large window sizes. It checks if the channel is idle and then the Backoff is reduced one unit. Else, it sends the packets if the channel is idle for more time until the Backoff time has a zero value as examined in [10, 11].

This mechanism leads to decrease the expected wait time by reaching a large window size for a specified node, allowing this node to access the shared medium and thus to avoid increasing in channel idle times. The Fibonacci series has a valuable characteristic to provide the precise value which obtained by the ratio of successive terms in the Fibonacci series [15].

The Backoff algorithms were proposed in order to avoid the collision and to resolve contention among different nodes as well as to improve the network resources utilization. Once there is a collision, retransmission delay occurs which indicates the nodes' needs to make a distinction in terms of time for a period time. Backoff time value is chosen randomly from bounded contention window which varies according to the consequence of the latest tries of the transmission based on the number of active nodes and traffic load in the network.

The Backoff algorithm takes place and it is performed in each of the next cases:

- If the channel is busy before the first transmission.
- After each attempt of retransmission
- After the node is successfully transmits a package.

Most of proposed Backoff algorithms behave unsatisfactory in case of failure to transmit rapidly the data in case of increasing the contention window; this clearly appeared in binary exponential Backoff algorithm. Other algorithms do not enable the node with enough time before retransmitting, which results in more power usage and more network overhead. For example, the exponential increment of BEB algorithm which is used in standard IEEE 802.11 MAC does not achieve the best performance due to large Contention Window (CW) gaps produced. Another example is a linear increment of Linear Multiplicative Increase and Linear

Decrease (LMILD) [9]; it does not allow a sufficient Backoff time before data retransmission.

The Backoff Algorithms are divided into two categories [1, 10]:

- Static Backoff Algorithms

In this category, Backoff algorithms proposed using a fixed Backoff wait time period which all nodes have a fixed Backoff waiting time based on the equation 2.

$$\text{Backoff Timer}=I, \text{ where } I \text{ is a fixed integer} \quad (2)$$

- Dynamic Backoff Algorithms

In this category, Backoff algorithms proposed using a variant Backoff wait time period by randomly choosing the Backoff timer value depending on equations 3 and 4 [11]:

$$CW_{new} \begin{cases} \text{Max}(f(CW), (Cwmax), \text{ after } I \text{ transmission} \\ \text{Min}(g(CW), (Cwmin), \text{ after a collision.} \end{cases} \quad (3)$$

$$\text{BackoffTimer} = b, b \text{ is random integer} \quad (4)$$

The rest of this section presents the most related work to represents the importance of pessimistic and Fibonacci Backoff algorithms.

S. Manaseer [1] introduces about some Backoff Mechanisms for Wireless Mobile ad-hoc Networks, focused on presenting and illustrating the importance of the two proposed Backoff algorithms [1] called Pessimistic Linear-Exponential, and Optimistic Linear-Exponential. Experimental results demonstrate that PLEB improves the network throughput, and reduces packet delay for large numbers of nodes and large network size with low mobility speed. On the other hand, OLEB has the same experimental results at high-traffic rates.

S. Manaseer, M. Ould-Khaoua and L. M Mackenzie in [13] introduce Fibonacci Backoff Algorithm for Mobile ad-hoc Networks. A Backoff Algorithm called Fibonacci Increment Backoff algorithm was proposed to reduce the differences among successive contention window sizes. Results from simulation experiments revealed that Fibonacci Increment Backoff algorithm achieves a higher throughput than the Binary Exponential Backoff algorithm used in MANETs. N. Song [14], enhanced the IEEE 802.11 distributed coordination function with an exponential increase and exponential decrease Backoff algorithm, whereas [14] also studied the effects of increasing and decreasing the waiting time intervals using exponential Backoff algorithm. Results representing the exponential increase algorithm have a good results using the coordination function. J. Deng, et al. [9] proposed the Linear Multiplicative Increase and Linear Decrease (LMILD) Backoff algorithm. This algorithm had shown a best performance and aims to achieve best Contention Window (CW) size. If failure transmission occurs it uses a factor multiplicative and linear increment; firstly, multiplicative the contention window by colliding nodes when there is a collision, other nodes hearing this collision make a linear increment to their contention window. On the other hand all nodes decrease their contention windows linearly when there is a transmission success.

V. Bharghavan, et al. [10] proposed Multiplicative Increase and a Linear Decrease (MILD) Backoff algorithm. This algorithm aims to solve the unfairness problem by reducing the probability of successful users to access the channel. In this algorithm the contention window size is incremented by a factor multiplication when a transmission failure occurs.

In our work, there are some important issues that should be taken into account when trying to design a Backoff algorithm that aims to improve the performance over the network such as determining the methods used to increase and decrease the CW and selecting suitable increase and decrease factors.

IV. PROPOSED PESSIMISTIC FIBONACCI BACKOFF (PFB) ALGORITHM

This section presents the proposed Pessimistic Fibonacci Backoff (PFB) algorithm which tries to organize node transmissions in time to avoid collisions. The PFB algorithm proposed is implemented to operate in collaboration with the IEEE 802.11 MAC protocol in order to reduce the collision in MANETs. This is achieved by incrementing the Backoff time by using a combination of Fibonacci, cubic and exponential increment behaviors.

In general, Backoff algorithms should use an appropriate increase for the contention window size in order to gain the best performance, because it should be incremented after each failure occurs in transmission. Some of the mentioned increment behaviors are appropriate when using a small or medium network dimensions but seems not enough good in large ones [1].

network high traffic load or a larger number of nodes located in a given network area.

The PFB algorithm proposed in next section aims to improve overall network performance mainly to achieve the best data delivery ratio with fewer routing load in the network. This algorithm uses a combination of Fibonacci, cubic and exponential increment behaviors.

Figure 2 presents the new proposed Backoff algorithm which referred as the Pessimistic Fibonacci Backoff (PFB). It assumes that there is congestion which is high enough and cannot be resolved in the near future.

PFB increases the contention window size exponentially and a transmission failure occurs to give a longer waiting time before retransmission, then PFB increases the timer cubically in order to avoid increasing Backoff extremely. After a fixed number of cubic increments PFB starts to increase the timer, using a Fibonacci series to achieve a less dramatic growth of the contention window size, allows nodes to perform more tries to access the channel.

```

PLEB Algorithm
0 Set Backoff timer to initial value
1 While BOT ≠ 0 do
2   For each time slot
3     If channel is idle then BOT = BOT-1
4 If channel is idle for more than IDFS then
5   Send
6 If send failure then
7   If NOB ≤ N then
8     CW = CW * 2
9     BOT = Rand (x);
10  Else
11    CW = CW + T
12    BOT = Rand (x);
13 Else
14   CW= Initial value
15   BOT = 0
16   Go to 1
17 Stop
Note : 1 ≤ X ≤ CW-1, NOB: Number of Backoffs, N,M: CW Threshold

```

Figure 1. Pessimistic Linear-Exponential Backoff (PLEB) Algorithm

The Pessimistic Linear-Exponential Backoff algorithm as in Figure1, assumes that congestion in the network will take more time to be resolved; it combines linear and exponential increment behaviors by using linear before using exponential at the first stages. PLEB algorithm adopts that a transmission failure is due to the network congestion resulted from the

```

PFB Algorithm
0 Set Backoff timer to initial value
1 While BOT ≠ 0 do
2   For each time slot
3     If channel is idle then BOT = BOT-1
4 If channel is idle for more than IDFS then
5   Send
6 If send failure then
7   If NOB ≤ N then
8     CW = CW * 2
9     BOT = Rand (x);
10  Else
11    If N < NOB < M
12      CW = CW^3
13      BOT = Rand (x);
14    Else
15      Use Fibonacci CW BOi+1=fib(I) --- CW = next fib
17 Else
18   CW= Initial value
19   BOT = 0
20   Go to 1
21 Stop
Note : 1 ≤ X ≤ CW-1, NOB: Number of Backoffs, N,M: CW

```

Figure 2. Pessimistic Fibonacci Backoff (PFB) Algorithm

When the congestion is found in the network the PFB adopts the corresponding transmission failure which is caused by a larger number of node's high traffic load within a limited area. When there is a transmission failure, PFB algorithm firstly increases the contention window size, mainly to give a long waiting time before starting the next transmission along the network paths. After that, PFB starts to increase the time using an exponential increment behavior in order to provide adequate values exploiting other paths, and then utilizing a cubically increasing. Finally, it uses the Fibonacci increment behavior in order to increase the CW size more exceptionally.

The intention of using these increment behaviors is to decrease the possibility in breaking paths which increased in a very sparse network, due to mobility and because there is single network path. This algorithm aims to achieve less

growth in the contention window size to allow the nodes to access the channel after incrementing Backoff time.

V. EXPERIMENTS SET-UP

This section explains the main methodology points used to direct this research, such as: implementing PFB, testing it and primary justification. Many parameters were used to measure the network efficiency such as delivery ratio, delay time, throughput, overhead, traffic load, and number of hubs. This work, uses three: end-to-end delay, normalization, and packet delivery Ratio.

A larger size of data was successfully received by nodes over the network due to the reduced amount of contention window size. During the simulation experiments the total number of nodes, the nodes pause time, and the maximum node speed are varied. Results are compared to both standard Backoff algorithm and Pessimistic Linear Exponential Backoff algorithm. NS-2 simulator [16] used to implement Pessimistic Fibonacci Backoff algorithm, and each figure represents the average of twenty autonomous runs. Our simulation is done using a personal laptop with 64 bits Ubuntu open-source operating system, the CPU is AMD phenom™ II N830 Triple-Core of speed 2.10GHz, and 8 GB of RAM.

TABLE I. LIST OF PARAMETERS

Parameter	Value
Nodes Number	25,50,75,100, 125
Area(x*y)	(900m*900m)
Connection Number	10
Pause time	0,150seconds
maximum speed	1,4,15 meters/second
Packet Size	512 bytes
Number of Packets	10, 20
Simulation Time	900 seconds
Bandwidth	2 Mbps

As mentioned above NS-2 simulator used and we implement proposed algorithm in it. We used the constant Bit Rate (CBR) traffic generator, therefore the sources and destinations are distributed in the network area randomly. The packet size is 512 bytes, sending 10 and 20 packets/second with a 10 number of connections. Nodes move at a random speed distributed in 900 meters X 900 meters mainly to stabilize the network, and to record more accurate results.

In this simulation, the value of proposed algorithm achieved improvement over standard and Pessimistic Fibonacci Backoff algorithm using equation 5.

$$Improvement = (Old Value - New value)/Old*100\% \quad (5)$$

VI. SIMULATION RESULTS AND ANALYSIS

The proposed topology simulations assume the following points:

- For the full length of simulation, nodes have sufficient power supply. At no point of the simulation lifetime, a node goes to offline because of lacking power.

- External network interference or noise does not exist. All the data that exist in the network is originated from within the network.
- Each node is equipped with a transmitter/receiver, or transceiver, IEEE 802.11 devices.
- The number of nodes over the network is constant for the length of simulation time. No nodes join nor leave the network for the duration of simulation.

Performing experiments and simulations, ideally is to achieve minimum delay, maximum packet delivery Ratio and minimum normalized routing load. Hence, the following factors will be used to measure the performance of the Pessimistic Fibonacci Backoff algorithm:

- End to End Delay [5]: is the average delay taken to transmit a packet through a network from source to destination. This delay may suffer from media access control retransmission delays and buffering the same time routing discovery.
- Packet delivery Ratio [6]: the number of received data packets by the destination nodes to the total number of transmitted data packets by the source node.
- Normalized routing load (overhead) [6]: the average number of sent routing packets to the total number of received packets.

Figure 3 represents the relation between end to end delay and network density of the STD (STD is the standard used algorithm which is Binary), PLEB and PFB scenarios, while each node has a movement speed up to 1 meter per second, and the transmission rate is 10 packets per second.

Figure 3 show that PFB algorithm achieves the lowest delay with specified area over all numbers of nodes. In case of dense environments (Maximum = 25 nodes), PFB outperforms STD and PLEB by 56.83% and 50.59%, respectively. In contrast, in sparse environments, PFB Outperforms STD by 40.41% and PLEB by 25.49%.

This is due to the fact that in the PFB, the number of increments and slow decrement behavior produced by exponential factor which leads to generate a longer Backoff values, is greater than PLEB, which leads to have a longer waiting times, and drive the contention window size to be larger instead of the need to achieve a minimum delay.

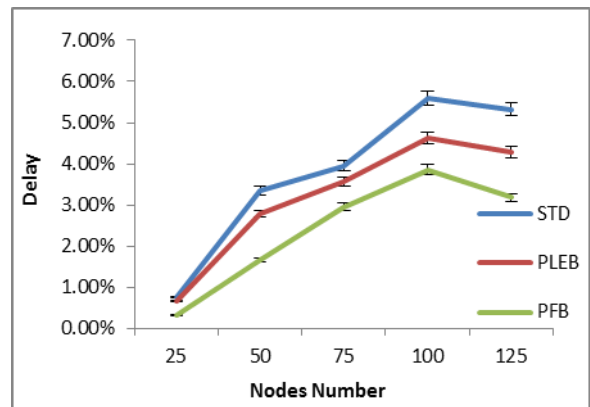


Figure 3: End To End Delay of STD, PLEB and PFB for Sending 10 packet/second, 0 second pause time, and 1 meter/second maximum speed.

Figure 4 shows end-to-end delay of the STD, PLEB and PFB scenarios over increasing number of nodes until reaching 125 nodes, 150 second pause time, while each node has a movement speed up to 4 meters per second, and the transmission rate is 10 packets per second.

This graph shows that PFB algorithm achieves the lowest delay with specified area over all numbers of nodes. In case of dense environments (Maximum = 25 nodes), PFB outperforms STD and PLEB by 13.61% and 10.67%, respectively. In contrast, for sparse environments PFB Outperforms STD by 25.88% and PLEB by 13.15%.

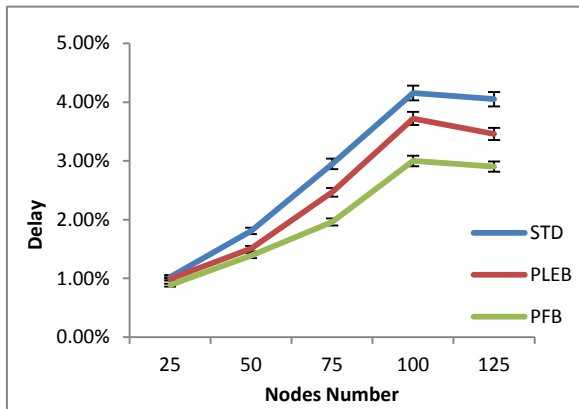


Figure 4: End To End Delay of STD, PLEB and PFB for Sending 10 packets/second, 150 second pause time, and 4 meters/second maximum speed.

Figure 5 shows end to end delay of the STD, PLEB and PFB scenarios over increasing number of nodes until reaching 125 nodes, while each node has a movement speed up to 1 meter per second, and the transmission rate is 20 packets per second. This graph shows that PFB algorithm achieves the lowest delay with specified area over all numbers of nodes. In case of dense environments (Maximum = 25 nodes), PFB outperforms STD and PLEB by 49.44% and 31.22%, respectively. On the other hand, for sparse environments PFB Outperforms STD by 35.99% and PLEB by 31.1%.

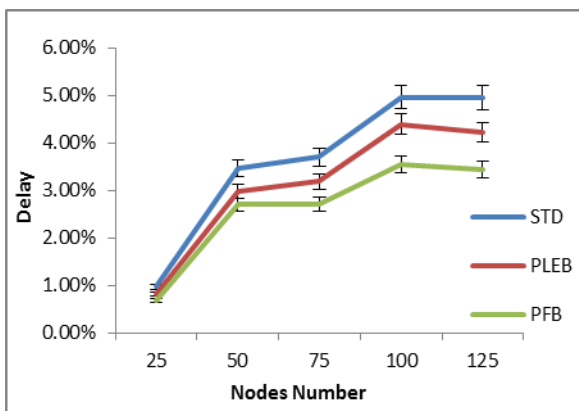


Figure 5: End To End Delay of STD, PLEB and PFB for Sending 20 packets/second, 0 second pause time, and 1 meters/second maximum speed.

Figure 6 shows end to end delay of the STD, PLEB and PFB scenarios over increasing number of nodes until reaching 125 nodes, while each node has a movement speed up to 4 meters per second, and the transmission rate is 20 packets per second.

In case of dense environments (Maximum = 25 nodes), PFB outperforms STD and PLEB by 68.26% and 49.33%, respectively. Oppositely, for sparse environments PFB Outperforms STD by 31.80% and PLEB by 21.80%. This graph shows that PFB algorithm achieves the lowest delay with specified area over all numbers of nodes.

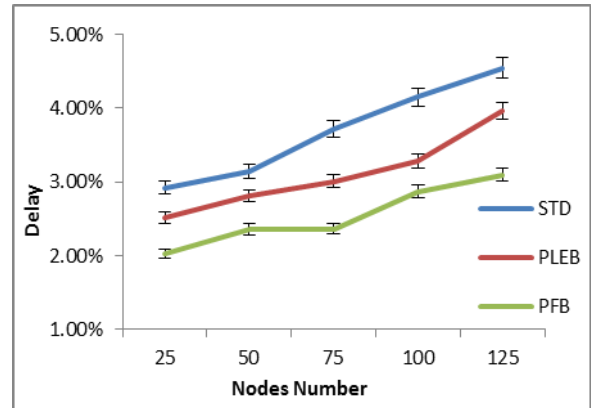


Figure 6: End To End Delay of STD, PLEB and PFB for Sending 20 packets/second, 0 second pause time, and 4 meters/second maximum speed.

Figure 7 shows end to end delay of the STD, PLEB and PFB scenarios over increasing number of nodes until reaching 125 nodes, 150 second pause time, while each node has a movement speed up to 15 meters per second, and the transmission rate is 20 packets per second. This graph shows that PFB algorithm achieves the lowest delay with specified area over all numbers of nodes.

In case of dense environments (Maximum = 25 nodes), PFB outperforms STD and PLEB by 20.83% and 12.10%, respectively. Inversely, on the Maximum number of nodes PFB Outperforms STD by 32.26% and PLEB by 21.19%.

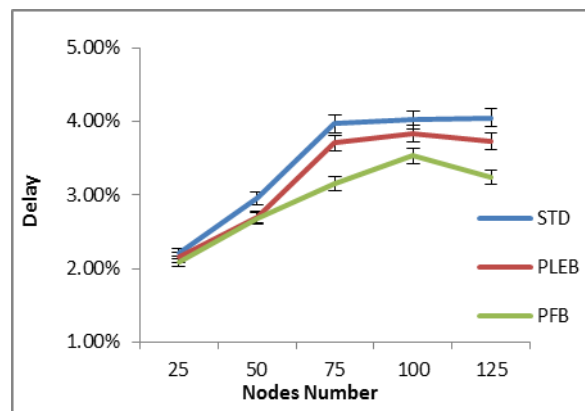


Figure 7: End To End Delay of STD, PLEB and PFB for Sending 20 packets/second, 150 second pause time, and 15 meters/second maximum speed.

Figure 8 shows packet delivery Ratio of the STD, PLEB and PFB scenarios over increasing number of nodes until reaching 125 nodes, while each node has a movement speed up to 1 meter per second, and the transmission rate is 10 packets per second. This graph shows that PFB algorithm delivers more packets than others with the specified area over the total number of nodes. When there are a few number of

nodes (maximum = 25 nodes), PFB outperforms STD and PLEB by 00.97% and 00.57%, respectively. Oppositely, on the maximum number of nodes PFB outperforms STD by 00.21% and PLEB by 00.14%.

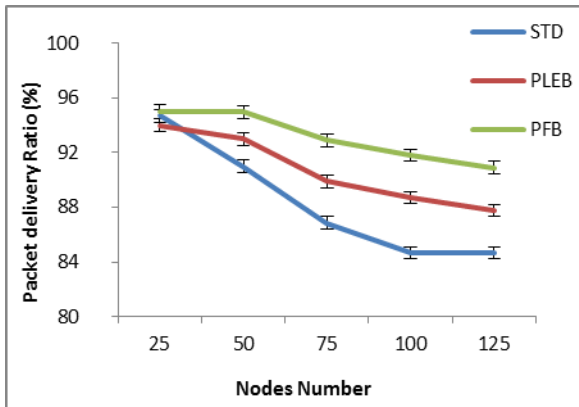


Figure 8: Packet delivery Ratio of STD, PLEB and PFB for Sending 10 packets/second, 0 second pause time, and 1 meters/second maximum speed.

Figure 9 shows the Packet Delivery Ratio of the STD, PLEB and PFB scenarios over increasing number of nodes until reaching 125 nodes, while each node has a movement speed up to 15 meters per second, and the transmission rate is 10 packets per second.

Figure 9 graph depicts that PFB algorithm delivers more packets than others with the specified area over all numbers of nodes. In case of dense environments (Maximum = 25 nodes), PFB outperforms STD and PLEB by 00.05% and 00.02%, respectively. Instead, on the maximum number of nodes PFB Outperforms STD by 00.11% and PLEB by 00.06%.

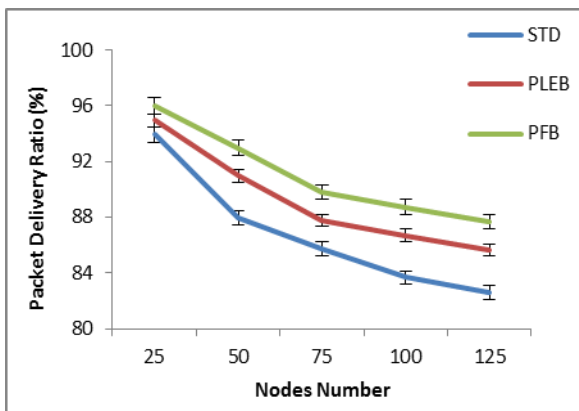


Figure 9: Packet delivery Ratio of STD, PLEB and PFB for Sending 10 packets/second, 0 second pause time, and 15 meters/second maximum speed.

Figure 10 shows packet delivery Ratio of the STD, PLEB and PFB scenarios over increasing number of nodes until reaching 125 nodes, 150 second pause time, while each node has a movement speed up to 15 meters per second, and the transmission rate is 10 packets per second.

This graph shows that the PFB algorithm delivers more packets than others with the specified area over all numbers of nodes. In the case of dense environments (Maximum = 25 nodes), PFB outperforms STD and PLEB by 00.02% and

00.05%, respectively. Oppositely, on the maximum number of nodes PFB Outperforms STD by 00.17% and PLEB by 00.11%.

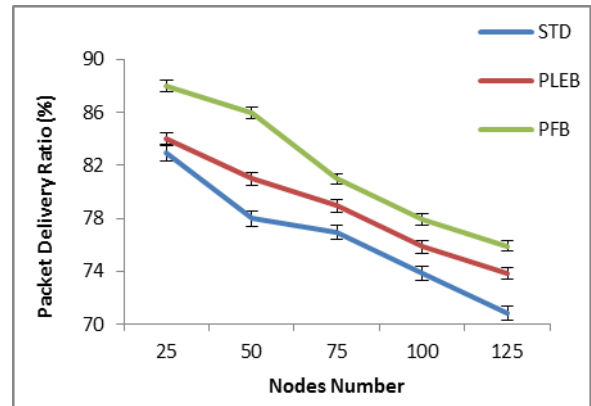


Figure 10: Packet delivery Ratio of STD, PLEB and PFB for Sending 10 packets/second, 150 second pause time, and 15 meters/second maximum speed.

VII. CONCLUSION AND FUTURE WORK

This research proposes the PFB algorithm designed which is compared in contrast to the STD and PLEB algorithms. The results show that the new proposed algorithm outperforms the other compared ones, in terms of packet delivery fraction, end-to-end Delay and normalized routing load. Results also show the effect of node speed in all three algorithms.

When the nodes speed is high, the delivery ratio is lower than, when the motion experienced by the node when having slow movements.

Results show the network density variations in terms of the parameterized measurements, where sparse areas have the highest value compared to the dense ones. Results also show that for different approaches, as the node speed increases, the number of successfully delivered packet, increases.

Moreover, this study highlights the importance of developing a new Backoff algorithm that can dynamically adjust the Backoff waiting time, and takes into confederation the increased contention window size that has to be progressively reduced.

Another improving area includes investigating the effect of node's transmission ranges. The PFB algorithm proposed can be used with any Medium Access Control protocol, to achieve uniform distribution using Fibonacci series by reducing the increment factor for large contention window sizes.

The results extracted by conducting simulation experiments show that PFB algorithm achieved better performance than PLEB by a percentage that ranges between 0.01% up to 74% in the cases studied for different network densities and mobility states for the end-to-end delay, the packet delivery Ratio, and the normalized routing load.

In a number of limited cases the PFB algorithm performance was not the optimal, which means that in any of these cases the devices' performance cannot reach peak compared to other established algorithms.

In order to find the most optimal network performance we can test different parameters, increment behaviors and distinct environment conditions.

A future work can merge more Backoff algorithms with different distributions to examine different variations of the pessimistic Fibonacci Backoff algorithm, including more parameters for measuring and validating the efficiency. Finally one of our priorities in future aims of the current research is to evaluate the proposed scheme under real-time parameterizations and conditions using WSN and more specifically the MICA2Dot motes. This will enable in real time the evaluation and efficiency of the proposed scheme under real-time conditions and experimentation.

VIII. REFERENCES

- [1] S. Manaseer. "On Backoff Mechanisms for Wireless Mobile Ad Hoc Networks". Thesis Submitted By For The Degree of Doctor of Philosophy. The Faculty of Information and Mathematical Sciences. University of Glasgow, 2009, PP 1-156.
- [2] M. BaniYassein, M. Khaoua, L.M. Mackenzie, S. Papanastasiou. "Performance Analysis of Adjusted Probabilistic Broadcasting in Mobile Ad Hoc Networks". International Journal of Wireless Information Networks, Springer Netherlands, Mar 2006, pp. 1-14.
- [3] H. Wu, Y. Pan. "Medium Access Control in Wireless Networks (Wireless Networks and Mobile Computing)". Nova science publishers, 2008.
- [4] K. Kalepu, S. Mehra, C. Yu. "Experiment and Evaluation of a Mobile Ad Hoc Network with AODV Routing protocol". MCRL (Mobile Computing Research Lab.), ECE Dept., Cleveland State University, Oct. 2002.
- [5] B. Williams, T. Camp. "Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks". ACM international symposium on Mobile ad hoc networking computing (2002), ACM Press, 2002, pp. 194-205.
- [6] Y. Tseng, S. Ni, Y. Chen, J. Sheu. "The Broadcast Storm Problem in a Mobile Ad Hoc Network. Wireless Networks". IEEE, presented at Wireless Networks, 2002, pp. 153-167.
- [7] M. Masadeh, S. Manaseer. "Pessimistic Backoff for Mobile Ad hoc Networks". IEEE, in proceeding of the 4th international conference on information technology ICIT, Jordan, 2009, pp. 1-10.
- [8] J.Al-Karaki, A.Kamal. "Routing Techniques in Wireless Sensor Networks: A Survey". IEEE communications, Volume 11, No. 6, 2004, pp. 6--28.
- [9] A. Escolà, J. Cantero. "Development of a wireless sensor network with 6LoWPAN support". Master thesis in Science and Telecommunication Engineering & Management, 2009.
- [10] C. Liu, J. Kaiser. "A Survey of Mobile Ad Hoc network Routing Protocols". TR-4, MINEMA, University of Magdeburg, 2005.
- [11] M. Kumar, T.V. Kumar, M. Hanumanthappa, E. Geetha. "Secure Mobile Based Voting System". Proceedings of 6th International Conference on E-Governance, IIT Delhi, 2008.
- [12] S. Eichler, C. Roman. "Challenges of Secure Routing in MANETs: A Simulative Approach using AODV-SEC". In Proceedings of the 3rd IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS), Vancouver, Canada, 2006, pp. 481 – 484.
- [13] S. Manaseer, M. Ould-Khaoua, L. M Mackenzie. "Fibonacci Backoff Algorithm for Mobile Ad Hoc Networks". DCS Technical Report series, Department of Computing Science, University of Glasgow, 2006, pp. 1-6.
- [14] N. Song. "Enhancement of IEEE 802.11 distributed coordination function with exponential increase exponential decrease Backoff algorithm". IEEE 57th Semiannual Vehicular Technology Conference, Vol.4, USA, 2003; pp. 2775-2778.
- [15] S. Manaseer, M. Ould-Khaoua, L. Mackenzie. "Fibonacci Increment Backoff Algorithm for MAC Protocol in Mobile Ad Hoc Networks". 7th Annual Postgraduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting, PGNET, Liverpool, UK, 2006, pp. 103-109.
- [16] NS-2 Simulator, at <http://www.isi.edu/nsnam/ns/>.