

# A New Fibonacci Backoff Method for Congestion Control in Wireless Sensor Network

Elamathy Selvaraju and Ravi Sundaram

**Abstract** Congestion control is one of the predominant challenges in wireless sensor network. It has a great impact on the parameters of quality of service such as end-to-end transmission delay, packet delivery ratio (PDR), and energy consumption in wireless sensor networks. Typical congestion control schemes include binary exponential backoff (BEB) or truncated binary exponential backoff (TBEB). This involves retransmission of frames in carrier sense multiple access with collision avoidance (CSMA/CA) and carrier sense multiple access with collision detection (CSMA/CD) networks. In this work, a new backoff strategy Fibonacci Backoff Algorithm (FBA) for congestion control is proposed. Each node is allocated a wait period as an incremental period. The simulated results show its better performance by decreasing the possibility of two or more nodes choosing the same backoff period, thereby decreasing the probability of collision and energy used for retransmission.

**Keywords** Congestion control · Fibonacci backoff algorithm · Performance analysis · Optimal scheduling

## 1 Introduction

In wireless sensor networks, nodes are deployed in large quantities where sensors can organize themselves to an ad hoc multipath network environment for communication between them. Each node is aware of its neighborhood nodes. Network protocol holds a strict layered structure and implements congestion control,

---

E. Selvaraju (✉) · R. Sundaram  
ECE Department, Dr. M.G.R. Educational and Research Institute University,  
Chennai, India  
e-mail: elamathy.j@gmail.com

R. Sundaram  
e-mail: ravi\_mls@yahoo.com

scheduling, and routing protocols. However, the time-varying nature with different channel conditions proposes a significant challenge to accomplish the above goals. Traffic engineering, end-to-end rate adaptation, and transport layer signaling have been widely developed to prevent the network congestion.

Shashi Kiran et al. [1] proposed an algorithm to trade-off between the latency and real-time capacity to adapt the transmission schedule in accordance with the addition, removal, and changes in dynamic queries. For fixed transmission schedules, the proposed method has the ability to overcome the changes in workload more effectively than the time division multiple access. A fair end-to-end window-based congestion control protocol for a packet-switched network is demonstrated in Shashi Kiran et al. [2]. with first come-first served (FCFS) routers. Only the end host information is used by the protocol. The network considered is a multiclass fluid model. The convergence function of the protocol is proved using a Lyapunov function.

A joint optimal design of cross-layer congestion control (CCC) for wireless ad hoc network is proposed in Chen et al. [3]. The rate and scheduling constraint is formulated using multicommodity, flow variables fixed channels in networks. The resource allocation problem is addressed through natural decomposition method. Congestion control, routing, and scheduling are the subdivisions. The dual algorithm is used to handle the time-varying channels and adaptive multirate devices. The stability of the resulting system is established. The performance is characterized with respect to an ideal reference system.

The resource like time slots, frequency, power, etc., at a base station to the other network node where each flow is intended for a different receiver. A combination of queue-based scheduling at the base station and the congestion control is implemented at the base station or at the end user node leading to the fair resource allocation and provide queue-length stability [4]. A redesign framework for fluid-flow models for network congestion control is proposed [5]. An extra dynamics are introduced using augmented Lagrangian method to improve the performance of the network.

## 2 Congestion in Wireless Sensor Networks

Among the various reasons for the congestion in a wireless sensor network, overloading of data, coherent nature of traffic to the base station node in the network due to many-to-one topology, ingestion of sensory data into the wireless networks plays a key role for the predominant congestion. Congestion affects the continuous flow of data, loss of information, delay in the arrival of data to the destination node, unwanted consumption of energy at each node. Basically, there are two types of congestion, node level and link level.

**Node level congestion**

The node level congestion is found to be common in conventional network. The common reason for its occurrence is the overflow of data at node. Its major effects are packet loss and increased queuing delay.

**Link level congestion**

It occurs in wireless sensor network, where nodes transmit the data packets at the same time. It significantly affects the link utilization and overall throughput with an increase in delay time and energy utilization.

**3 Fibonacci Backoff Algorithm**

The nodes in the wireless sensor network are allowed to wait for a period which is formulated from the mathematical Fibonacci series. The wait period formulated is in incremental period. The subsequent numbers are generated directly adding the previous two numbers according to the given formulation.

$$F(x) = f(x - 2) + f(x - 1), \text{ where } x \geq 0.$$

$$F(0) = 0, f(1) = 1.$$

There are two methods used to prevent the congestion. The possibility of two or more nodes in the network to transmit the data at the same time increases the possibility of congestion among them.

**3.1 Method I**

Nodes are allowed to choose a random backoff period as the below mentioned range.

$$0 - 2^n - 1,$$

where  $n$ —number of collisions.

For the first collision, each sender node has to wait for a 0 or 1 time slot. The maximum backoff period for a node depends on the range of Fibonacci series which is chosen for the Fibonacci Backoff Algorithm (Table 1).

From this method, the retransmission attempts can be analyzed as follows,

Number of retransmission attempts	$\propto$ exponential increase in the number of delay possibility
-----------------------------------	---

**Table 1** Slot time generation

Number of collision	Slot time
1	0 or 1
2	0-3
3	0-7
4	0-256

Number of nodes considered in this work is 3, and the time slots considered is 10. The slotted time communication between the nodes and the station flow node is shown in Table 2.

### 3.2 Method II

The second method is based on the Fibonacci Backoff Algorithm (FBA), and a random Fibonacci number is generated based on the number of times the transmission failure has occurred. This method has two sections as node flow and code flow. There are three basic steps in transmission of data as follows,

1. The serial port is open,
2. Method of algorithm is selected either as ‘exponential’ or ‘fibonacci backoff’ algorithm, and
3. Data is sent.

**Table 2** Transmission status for nodes 1, 2, 3

Time slot	Node1	Node2	Node3
T1	X	Txd	X
T2	X	Txd	Txd
T3	Txd	X	Txd
T4	Txd	Txd	Txd
T5	Txd	Txd	Txd
T6	Txd	Txd	Txd
T7	Txd	Txd	Txd
T8	Txd	Txd	Txd
T9	Txd	Txd	Txd
T10	Txd	X	x

**Algorithm for node code flow**

Algorithm for slot time communication between the nodes is as follows,

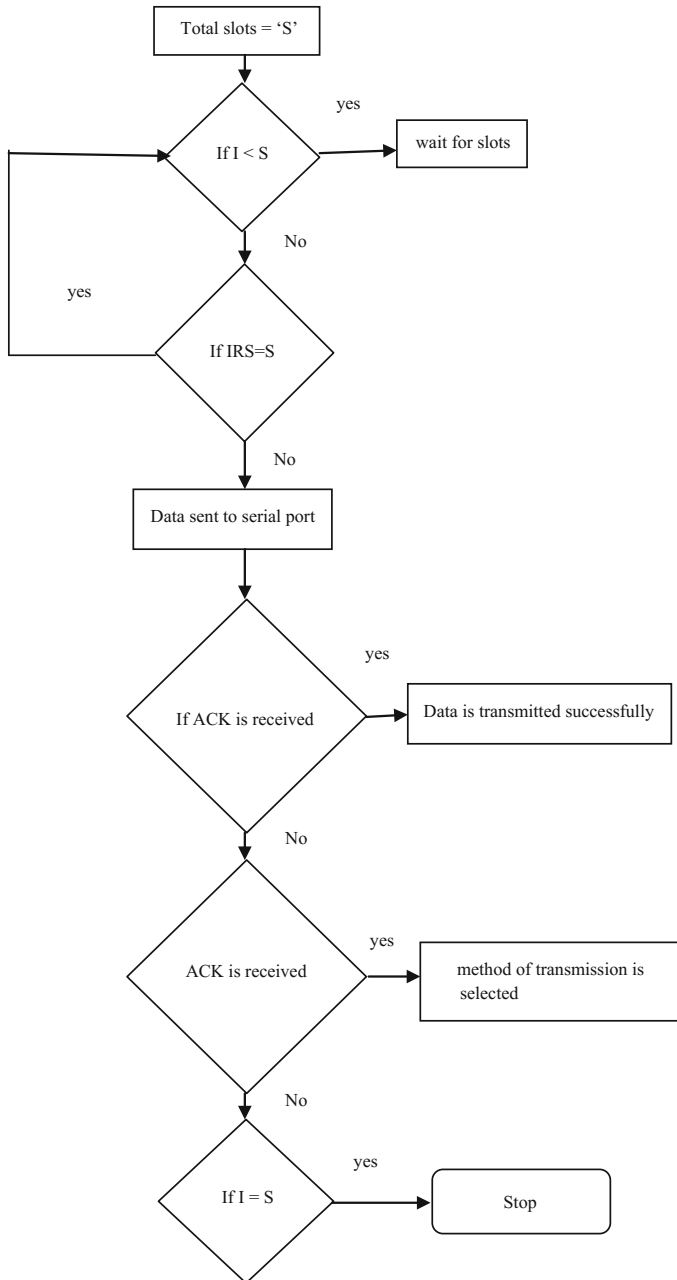
- Step 1: if index < TOTAL SLOTS.
- Step 2: wait for slots on serial port.
- Step 3: if received slot1 = 'S', go to step1.
- Step 4: data I MSG(n) is transmitted to the serial port, and node waits for acknowledgment or NACK.
- Step 5: data is sent successfully, once ACK msg is received.
- Step 6: if not, data is transmitted either into exponential or Fibonacci code as NACK is received.
- Step 7: random number is generated, and control is transmitted to the program.
- Step 8: the value returned represents the number of times, and the slot has to be skipped to avoid collision.
- Step 9: if index == TOTAL SLOTS, return.
- Step 10: else go to step 1.

**Station flow algorithm**

Algorithm for slot time communication for the station node is as follows,

- Step 1: serial port is initialized for communication with node1, node2, and hyper-terminal.
- Step 2: slot signal is sent to node1 and node2.
- Step 3: wait period is set, until the data is received from node1 and node2.
- Step 4: if node1 and node2 send data at the same time, ACK message is sent to both the nodes to signal the collision between them and discard of data.
- Step 5: if anyone of the node sends data during the slot time, ACK message is send to the node and received data is displaced.
- Step 6: if end of transmission is received from node1 and node2, station node is terminated else control is transferred to step2.

**Flow chart for node flow**



## 4 New Slot Allocation Table for Retransmission

### 4.1 First New Slot Generation for Retransmission for Node1, Node2, Node3

See Table 3.

### 4.2 First New Slot Generation for Retransmission for Node2, Node3

See Table 4.

### 4.3 New Slot Generation for Retransmission for Station Node

See Table 5.

**Table 3** First new slot allocation

Node	Collision occurred time slot	Newly allocated time slot
Node1	Slot 1 Slot 2	Slot 6 Slot 3
Node2	Slot 1 Slot 2 Slot 3 Slot 6	Slot 4 Slot 9 Slot 8 Slot 6
Node3	Slot 3 Slot 10	Slot 5 Slot 8

**Table 4** Second new slot allocation

Node	Collision occurred time slot	Newly allocated time slot
Node2	Slot 5 Slot 8	Slot 4 Slot 9
Node3	Slot 5 Slot 8	Slot 5 Slot 8

**Table 5** New slot allocation for station node

Time slot	Network status	Slot	Network status
1	Tx	1	Tx
2	Tx	2	Tx
3	X	3	Tx
4	Tx	4	Tx
5	Tx	5	X
6	Tx	6	Tx
7	Tx	7	Tx
8	Tx	8	X
9	Tx	9	Tx
10	Tx	10	Tx

### 5 Comparative Analysis Between BEB and FBA

In a wireless sensor network, distributed coordination function (DCF) is used for this best performance. When nodes want to access the channels, the backoff algorithm is executed while its backoff time greatly depends on the random number. Hence, it is difficult to predict exactly the backoff duration. In BEB, when data is sent and if acknowledgment (ACK) message is not received represents the occurrence of collision in the transmitted channel. The nodes involved increases its contention window size, and in case of successful transmission, the slot time is reduced by one. In this work, we have proposed Fibonacci Backoff Algorithm (FBA) which makes significant changes in the channel access mechanism. Two different scenarios have been introduced for the existing protocol.

In an actual BEB, when the number of collision is 3, the backoff time slot generated is obtained from  $2^n$ .

$$2^3 = 8(0 \text{ to } 7)$$

$$\text{Generated time slots} = 0, 1, 2, 3, 4, 5, 6, 7.$$

Whereas in Fibonacci Backoff Algorithm, the waiting time slot is generated using the formula,

$$F(x) = f(x - 2) + f(x)$$

For the considered condition of number of collision 3,

$$\begin{aligned} F(3) &= f(3 - 2) + f(3 - 1). \\ &= 1 + 2 = 3. \end{aligned}$$



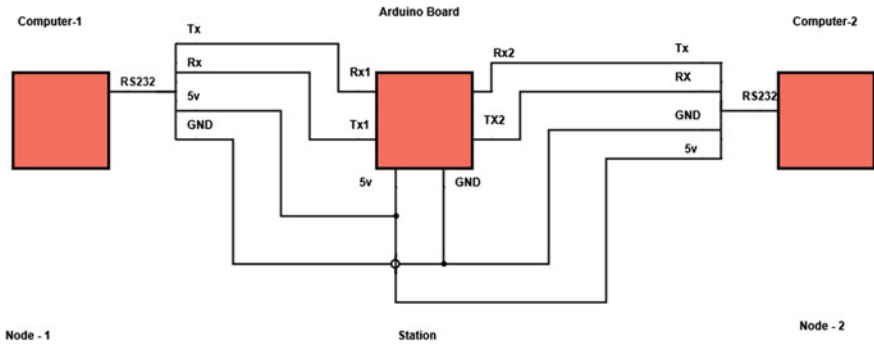


Fig. 1 Connection of nodes through Aurdino board

When  $n = 3$ , with the mathematical Fibonacci series generation, the waiting time slots generated are,

$$\text{Time slots} = (0, 1, 1, 2)$$

Number of time slots or the backoff/waiting period generated is 4, whereas in binary exponential backoff algorithm, the number of waiting slot generated is 8. Hence, the comparative study of BEB and FBA can be expressed as,

$$\text{Backoff period in BEB} = 2(\text{waiting time slot generated in FBA})$$

## 6 Hardware Setup

See Fig. 1.

# 7 Results

## Output Screen Shots

Node - 1 output:	Node – 2 Output:	Station output
<pre>Totalslots 6 select choice 1.exponential 2.fibonacci 1 choice= 1 index= 0 waiting for the slots, (s or n) slots = s data is empty  index= 1 waiting for the slots, (s or n) slots = s data is = hi  waiting for acknowledge,(s or n) ACK = n collision occured= 1 time exponential value 1.0 index= 1 waiting for the slots, (s or n) slots = s data is = hi  waiting for acknowledge,(s or n) ACK = n collision occured= 2 time exponential value 3.0 index= 1 waiting for the slots, (s or n) slots = s data is = hi  waiting for acknowledge,(s or n) ACK = n collision occured= 3 time exponential value 7.0 skip= 3 slots waiting for the slots, (s or n)</pre>	<pre>Totalslots 6 select choice 1.exponential 2.fibonacci 2 choice= 2 index= 0 waiting for the slots, (s or n) slots = s data is empty  index= 1 waiting for the slots, (s or n) slots = s data is = hi  waiting for acknowledge,(s or n) ACK = n collision occured= 1 time the nth number for that fibonacci sequence = 1 index= 1 waiting for the slots, (s or n) slots = s data is = hi  waiting for acknowledge,(s or n) ACK = n collision occured= 2 time the nth number for that fibonacci sequence = 1 index= 1 waiting for the slots, (s or n) slots = s data is = hi  waiting for acknowledge,(s or n) ACK = n collision occured= 3 time the nth number for that fibonacci sequence = 2 index= 1 waiting for the slots, (s or n)</pre>	<pre>----- 1.Press Any Key to Start Station -----  Start of Slot No Data Received from Node 1 or Node 2 Start of Slot Two nodes sent data in same slot Collision Occured  Start of Slot Two nodes sent data in same slot Collision Occured  Start of Slot Two nodes sent data in same slot Collision Occured  Start of Slot Data Received from Node 2 hi  Start of Slot No Data Received from Node 1 or Node 2 Start of Slot Data Received from Node 2 hello how r u  Start of Slot Data Received from Node 1 hi  Start of Slot Data Received from Node 2  good morning  Start of Slot</pre>

# 8 Conclusion

In this paper, a new Fibonacci Backoff Algorithm is proposed and implemented. A comparative analysis is done between the already existing binary exponential backoff (BEB) algorithm. The incremental nature of the Fibonacci Backoff Algorithm decreases the probability of occurrence of congestion by allocating the

waiting time slots based on the number of transmission failure occurrences. The newly proposed Fibonacci Backoff Algorithm (FBA) and the results obtained prove the effectiveness of by reducing the number of waiting time slots, and the waiting period is reduced to half than the BEB algorithm. Hence, the network throughput increases by congestion prevention and decreased energy utilization.

## References

1. Shashi Kiran, B.S., R.L. Pavan Shree, Sanjay K. Nagendra. 2015. Query Scheduler for Secure Collision-Less Transmission in Wireless Sensor Networks. *International Journal of Modern trends in Engineering and Research*. ISSN 2349-9745.
2. Mo, Jeonghoon, and Jean Walrand. 2000. Fair End-to-End Window-Based Congestion Control. *IEEE/ACM Transactions on Networking* 8 (5).
3. Chen, Lijun, Steven H. Low, Mung Chiang, and John C. Doyle. 2006. Cross-layer Congestion Control, Routing and Scheduling Design in Ad Hoc Wireless Networks. In *IEEE Communications Society Subject Matter Experts for Publication in the Proceedings IEEE Infocom*.
4. Zhou, Anfu, Min Liu, Zhongcheng Li, Eryk Dutkiewicz. 2015. Joint Traffic Splitting, Rate Control, Routing and Scheduling Algorithm for Maximizing Network Utility in Wireless Mesh Networks. *IEEE Transactions on Vehicular Technology*. doi:10.1109/TVT.2015.2427091.
5. Zhang, Xuan, and Antonis Papachristodoulou. 2015. Improving the Performance of Network Congestion Control Algorithms. *IEEE Transactions on Automatic Control* 60 (2).