Performance of Sensor MAC Schemes and Prediction Model For Bursty Traffic
Final Report

Akshita Maradapu Vera Venkata Sai and Naveen Chalicheemala

Abstract— **In an Ethernet LAN, computers and routers are interconnected by a shared medium called Ethernet.Thus, all the computers and the routers need to compete to be able to transmit data packets among them.The problem with the shared medium is allocating the shared resources among the hosts.This issue is resolved by using certain protocols called MAC protocols.But, the problem with MAC protocols is that they were initially designed for wired networks, this which have point to point connections and implementing them on a wireless networks where the nodes communicate via sensors makes the protocols function inefficiently.**

Most of the MAC protocols proposed work efficiently on a static network and do not support networks where static and mobile nodes co-exist and they do not support the dynamic and time varying traffic that occurs in a WSN.

Most of protocols go with random hop selection for a mobile node , where the mobile node responds and sends data to the node that sends the first ACK packet, even if the static node is many hops away.This approach may not be efficient in most cases as it increases time delay for the overall network. So, a protocol must be designed or established that allows for effective and secure communication between the mobile nodes and the static nodes while reducing the overall time delay and energy consumption in the network.

I. Introduction

Internet of Things is a network of computing devices , mechanical and digital machines that are provided with unique identifiers and their ability to transfer data over a network without requiring any interaction.These devices can communicate with each other over the network infrastructure and this is possible due to the presence of embedded and the external sensors.

A Wireless Sensor Network(WSN) is a network of spatially distributed devices using sensors to monitor physical and environmental conditions.This provides a gateway between the wired nodes and the distributed nodes.

MAC protocols are in charge of the communication between nodes and handles all operations related to packet transmission and reception. In IoT and WSNs , MAC layer is also responsible for switching the radio device ON and OFF at periodic intervals.

Even if there are many mobile networks present today, most of their topologies are fixed or static and the next hop of the nodes changes with changes in the physical layer. A large number of MAC protocols present today do not take the mobility of the nodes and the variable traffic in the network into consideration.

The original Globecomm.paper proposes a MAC Scheme called MobIQ which achieves low-delay mobile-to-static communication. This particular scheme helps in efficient and effective neighbourhood discovery and supports dynamic traffic.It is also said to handle the connection fluctuations and disconnections that may occur often in a wireless network.It also overcomes the hidden terminal and network contention by using a Contention Avoidance Algorithm. The goals achieved by the paper are: 1.Introduction of MobIQ Scheme that allows selective and low-delay mobile to static communication and this protocol belongs to the family of preamble sampling MAC protocols

2.Introduces Contention Avoidance Algorithm to leverage channel contention and hidden terminal problem.

3.The network is generated over COOJA and the protocol is compared with other MAC protocols like Me-Contiki MAC , Mox-MAC and MOBINET.

The kind of network on which the protocol is being tested is a hybrid network, where the mobile nodes do not perform any routing functions due to their velocity and sleep duration and these nodes communicate with the static wireless sensor nodes.The aim of the mobile nodes is to set up point to point links for data transmission and dequeue their transmission once they identify or discover any static node.The static nodes are assumed to have complete routing information at any point and this information is made available to the mobile nodes in the form of ACK packets so that they can make the decision of selecting the next hop.

A. Random Hop Selection

In this the mobile node randomly choses a static node as the next hop. Though , this may provide the best hop in few cases , in most of the times the hops thus selected may increase the end-to-end delay and the energy consumption because of longer routing paths. But, in MobIQ the nodes perform Neighbourhood discovery where the mobile nodes send control packets during the preamble period to all the neighbour static nodes via an any-cast transmission. The receivers of the control packets responds to the mobile nodes via an Acknowledgement packet and depending on the information retrieved from these acknowledgments the temporary next hop is selected.

B. Contention Avoidance Algorithm

This is introduced in the protocol to avoid the hidden terminal problem and the channel contention. For this contention avoidance algorithm , the queue length of the data is incorporated into every MobIQ data packet.The other mobile nodes listens to this information and calculates the time required for the transmission and accordingly calculate their own sleep duration.The mobile nodes will then turn their radio OFF for that time duration in over to save energy.

II. Scope and Purpose of the Project

Bursty traffic is quite common when it comes to wireless sensor networks.Bursty traffic is high bandwidth transmission over a short period of time. The problems with bursty traffic are congestion, bottlenecks in the network , increased time delay and packet losses.This issue can be resolved by predicting when the bursty traffic occurs in the network. This can be done by using a combination of machine learning algorithms and prediction models as described in [1]

Packet losses are a major issue when it comes to wireless networks.Lost packets or data may lead to broken content in case of images or vedios .There may be many factors that contribute to packet losses. One of the main reasons being connection timeout after the source has sent the data and the other being not having enough bandwidth for the data transfer. The paper only focused on the reducing the time delay and the energy consumption for data transfer. We will be looking into the packet loss aspect and try an implement a algorithm that looks into this problem and see how efficient the data transfer is and if possible even have a recovering plan for the lost data.

The MAC protocol used in the paper works efficiently on a static network, but gives comparatively better results when compared to the other protocols that were used for the comparison in the paper. We would like to compare the performance of the proposed protocol with those protocols that are specifically designed for the wireless sensor networks like SMAC , B-MAC , PW-MAC, Low-Energy Adaptive Clustering Hierarchy (LEACH).

When it comes to wireless sensor MAC protocols, there are two kinds in it.

1. Scheduling based protocols like LEACH and PEADAMAC 2.Contention based protocols like SMAC BMAC and PW-MAC For comparing the performance of to the common mac protocol , we have selected one protocol from each category.

S MAC: S-MAC operates by placing a node in a state that listens to the medium; if a node hears nothing it sends a SYNC packet with a schedule defining listen and sleep periods. All nodes hearing this packet will adopt the schedule. Nodes may adopt two or more schedules (if different neighbors have different schedules). Nodes keep tables with the schedules of their neighbors.

The advantages of SMAC are:

1. It reduces energy consumption.

2.The protocol adapts easily to changes in topology and has been tested in hardware.

3. Additionally, there is no need for a central entity or for tight synchronization.

The disadvantages of SMAC are:

1.Loose synchronisation has to be maintained for the schedules to work properly

2. Shifts in clocks causes unsynchronisation

3. RTS and CTS generate overhead

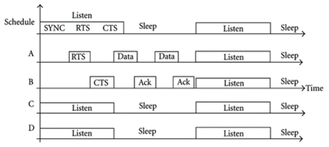


Fig. 1. Simple SMAC protocol working.

4. Idle listening still occurs **Low-Energy Adaptive Clustering Hierarchy (LEACH):** LEACH includes application, routing, MAC, and physical characteristics for communication in WSNs. A specific application considered is remote monitoring where data gathered by neighboring nodes may be redundant. LEACH assumes all nodes are synchronized, they can control their transmission power, and they can reach one base

station (BS, equivalent to the sink in other protocols) if needed.All the communications in the network are assumed to be only one hop communication. The protocol has two phases in it , the first phase called the initial setup , we look for selecting the cluster heads(CH) for which non-persistent CSMA is used.Node i selects itself as a ClusterHead .Once the cluster head is elected , it sends the advertisement packet. All the non-elected nodes receive this message and decide which cluster to join in the later round. In the second round, CH creates a TDMA :every node uses only its assigned time slot to send data to the CH and sleeps the rest of the time. CH sends the obtained data to the BS

The advantages of LEACH are:

1. Energy saving through sleeping
2. CH rotation extends the lifetime of the network

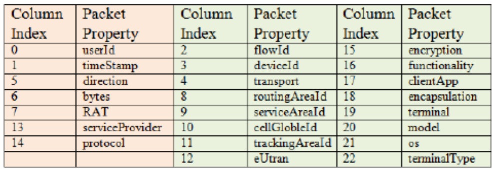
The disadvantages of SMAC are:

1. If the CH dies then the entire cluster becomes inactive
2. Assumes one hop communication between nodes
3. Requires tight synchronisation

III. Prediction Model for Bursty Traffic

I. Data Pre-processing

Data Collection: The model takes the data file as an input which consists of packet information from base stations. The data is in the form of array with each row standing for one packet and each column for properties of packet. This particular model has taken 23 different features out of which only seven are the most important features(userid, time stamp, direction, bytes, ram, etc) that are used for the actual model. The features of packets in data file looks as follows:



Column Index	Packet Property	Column Index	Packet Property	Column Index	Packet Property
0	userId	2	flowId	15	encryption
1	timeStamp	3	deviceId	16	functionality
5	direction	4	transport	17	clientApp
6	bytes	8	routingAreaId	18	encapsulation
7	RAT	9	serviceAreaId	19	terminal
13	serviceProvider	10	cellGlobleId	20	model
14	protocol	11	trackingAreaId	21	os
		12	eUtran	22	terminalType

Fig. 2. Packet features.

Defining the burst: In order to define the burst, the value of X should be defined where ”X” is the maximum inter-packet gap two packets can have in order to belong to the same burst. If the value of X is small, then the number of bursts will be more when compared to large value of X. In order to find the good value for X, the number of bursts should be observed for values between 0.1 and 1.

Generating Burst Data: Once the bursts have been calculated, the packets are grouped on the basis of the user Id and the arrival time. Firstly, the data is grouped according to the user Id and the packets of one user are sorted by their time stamps. Each user’s packets are grouped according to the arrival time and packets whose inter-packet gaps are lesser than X are grouped under one burst. The generated burst data will now have its own set of features (user Id, burst Id, start Time, dur Time etc) and then the burst data is processed either by multi-core processing or big data processing using Apache Spark(using Map Reduce Techniques).

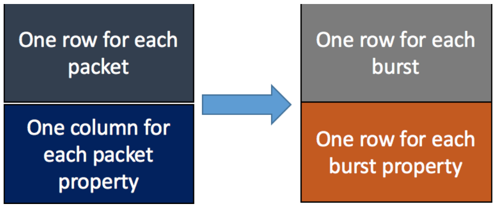

One row for each packet
One column for each packet property
One row for each burst
One row for each burst property

Fig. 3. Packet features.

.

II. Machine learning model: The model has [X,y], where X is the feature matrix and y is the label vector an each row is an input to the machine learning algorithm and the corresponding number in vector y is the output of the particular example. The data set is split into two sets: training data set and testing data set. Training data set is used to build the classifier and testing data is used to test the trained classifier.

1.Features of prediction :

1.1 Feature Extraction : The objective of feature extraction would be to extract the useful data from the raw data. The raw data here would be the burst properties which are represented by the columns in burst file and the prediction here is based on the idea that there is a pattern in the history of bursts of a user. Three burst features are predicted using this model. They are burst volume, burst duration and burst gap. Burst Volume is the sum of bytes of all the packets in the same burst. Burst duration time is the time interval between the first packet and last packet in the same burst. And burst gap is the time interval between the first packet in the current burst and the last packet in the previous burst. The example to find how the features are extracted for volume prediction is as follows: avg-volume is the average volume of N bursts. The average number can be obtained by dividing the sum of all bytes of N bursts by N.

1.2 Feature Construction: After the features have been extracted from the burst data a few features are generated for the ML model , these features are needed to indicate the structure of the data format as well as the trend of bursts.

1.3 N and n : Here N bursts are considered for each user , therefore N becomes the maximum number of bursts to create each feature and an additional 'n' bursts will be used to indicate the recent traffic status of the user , thus , 'n' is the minimum number of bursts used to create a feature. Therefore, for this model 'N' is the upper bound of feature engineering while 'n' is the lower bound. Selecting the value of N and n depends on how long the time can be a history of user and how short a time is capable of indicating the recent status. By analyzing the inter-arrival time of the bursts, the number of bursts can be related to the time.

2.Adding labels for supervised learning: For each prediction there is a threshold. If the predicted property in that burst is more than the threshold then the classifier sets the value to 1 else it is 0. The main goal of burst prediction is to allocate resources in an efficient manner. The bursts with value among the majority have less meaning to this prediction. Instead, the bursts with large values such as big volume are more important when making resource allocation decisions.

Set Split: This involves dividing the data into different types. For the current model the data is divided into two types : Training Data and Test Data. In most of the cases the split is done randomly but in this as timestamps are involved random splitting of data might cause information leakage. In order to avoid this leakage the data splitting is done in two ways :

1. Horizontal split: Here , the data is split into training data and testing data according to time stamp and this is done by introducing a time stamp breakpoint.

2.Vertical split: Here the data is split into sets according to the user Id. In this way all the bursts of one user will be either in the training set or the test set , thus there wont be any past information of the user's burst data which prevents information leakage in a way.Here simply , the split percentage will tell what percentage of users will be in the training set.

III. Implementation of the model:

1.Feature set:

1.1 Burst Volume Prediction:

FeatureMatrix = [perc all vol, perc rec vol, perc vol change, currentRAT, currentV OL, currentSize, currentPro, currentService]

currentRAT: The RAT of the latest packet in the latest burst.

currentVOL: the number of bytes in the latest burst.

currentSize: Number of packets in the latest burst.

currentPro:protocol of the latest packet in the latest burst

currentService:the service of the latest packet in the latest burst.

1.2 Burst Duration Tine Prediction:

FeatureMatrix = [perc_all_dur, perc_rec_dur, perc_dur_change, avg_rec_dur, avg_rec_gap, avg_rec_size, currentRAT, currentDur, currentSize, currentPro, currentService, currentGap, pre avg dur, pre cur dur]

avg_rec_dur average duration of recent bursts

avg_rec_gap average gap of recent bursts

avg_rec_size average size of recent bursts

currentGap gap of the recent burst

pre_avg_dur : pre_avg_dur = avg_rec_gap x (avg_rec_size 1)

pre_cur_dur : pre_cur_dur = currentGap x (currentSize 1)

IV. Tools and Network Simulation

A. NS2:

NS2 is an open-source simulation tool that runs on Linux. It is a discreet event simulator targeted at networking research and provides substantial support for simulation of routing, multicast protocols and IP protocols, such as UDP, TCP, RTP and SRM over wired and wireless (local and satellite) networks. It has many advantages that make it a useful tool, such as support for multiple protocols and the capability of graphically detailing network traffic. Additionally, NS2 supports several algorithms in routing and queuing. LAN routing and broadcasts are part of routing algorithms. Queuing algorithms include fair queuing, deficit round-robin and FIFO.

We have simulated a Wireless Network with eight nodes in it for the initial setup.

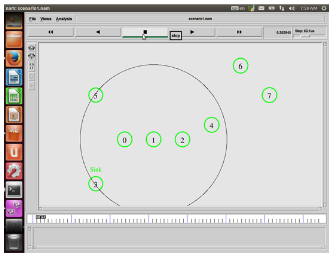


Fig. 4. Simple Wireless Network.

B. Gnuplot

gnuplot is a command-line program that can generate two- and three-dimensional plots of functions, data, and data fits. It is frequently used for publication-quality graphics as well as education. The program runs on all major computers and operating systems (Linux, Unix, Microsoft Windows, macOS, and others).

V. Network Configuration

1.Number of nodes: 100
2.Dimensions : 800 by 800 units

VI. Result for SMAC

The protocol has been tested over a wireless sensor networks with 100 nodes.

A. 1.Measurement of Energy Consumption

Once the .tcl script is run , a .trace file is created which contains the data for energy measurement . Using the data of the trace file we plot the graph in gnuplot.From the graph it is evident that the energy consumption stays constant for the SMAC protocol as it can easily adjust to the topology whereas, with the basic IEEE 802.11 the energy consumption increases linearly with time.

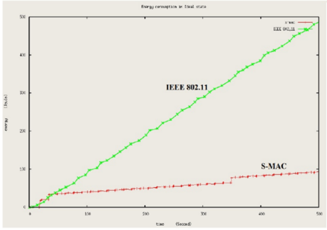


Fig. 5. Energy Consmption vs Time.

B. Energy consumption with delay

We have tested the protocol for two duty cycles-10 % and 20%, where duty cycle is the ratio of the awake time to the the listening time . From the graph it is evident that the energy consumption is more when the duty cycle is 20% but the spike in energy is not that high

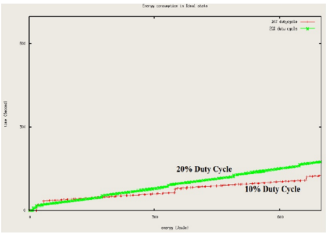


Fig. 6. Energy Consmption vs Duty cycle

C. Energy consumption with traffic variation

In order to see the energy consumption, we have tested it for two different packet arrival intervals. When the interval is set to 0.001s we have high traffic and when the interval is set to 10.0s we have the low traffic. From the graph SMAC's energy consumption stays constant as it adjusts to topology easily while with the IEEE 802.11 there is a sudden spike in the energy consumption.

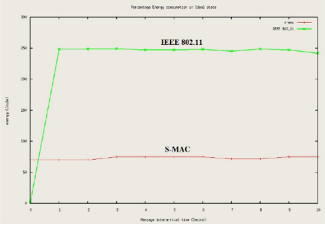


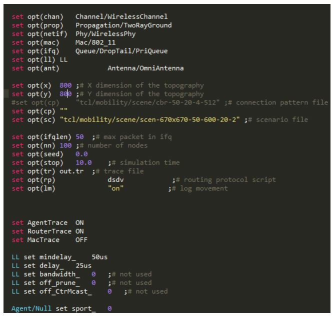
Fig. 7. Energy Consmption vs Traffic.

VII. Result for LEACH

With this protocol we have 100 nodes named from 0 to 99 and the last node is said to be the base station.The nodes are placed randomly over a area of 800 by 800 units using a random number generator.

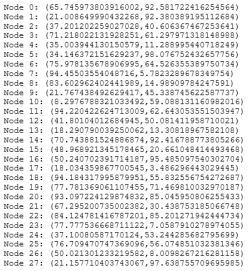
A. Energy Consumption

At the end of the run of the code, we get the total amount of time network has been running and gives the amount of energy consumed by each node of the network which here is fixed.The intial run of the code, gives the cluster heads in the network for that round and these cluster heads change in the next round.



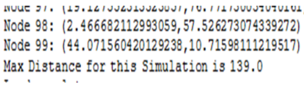
```
set opt(chan)   Channel/WirelessChannel
set opt(prop)   Propagation/TwoRayGround
set opt(netif)  Phy/WirelessPhy
set opt(mac)    Mac/802_11
set opt(ifq)    Queue/DropTail/PriQueue
set opt(ll) LL
set opt(ant)            Antenna/OmniAntenna

set opt(x)  800 ;# X dimension of the topography
set opt(y)  800 ;# Y dimension of the topography
#set opt(cp)    "tcl/mobility/scene/cbr-50-20-4-512" ;# connection pattern file
set opt(cp) ""
set opt(sc) "tcl/mobility/scene/scen-670x670-50-600-20-2" ;# scenario file

set opt(ifqlen) 50  ;# max packet in ifq
set opt(nn) 100 ;# number of nodes
set opt(seed)   0.0
set opt(stop)   10.0    ;# simulation time
set opt(tr) out.tr  ;# trace file
set opt(rp)             dsdv            ;# routing protocol script
set opt(lm)             "on"           ;# log movement



set AgentTrace  ON
set RouterTrace ON
set MacTrace    OFF

LL set mindelay_    50us
LL set delay_   25us
LL set bandwidth_   0   ;# not used
LL set off_prune_   0   ;# not used
LL set off_CtrMcast_    0   ;# not used

Agent/Null set sport_   0
```

Fig. 8. Initial set up.



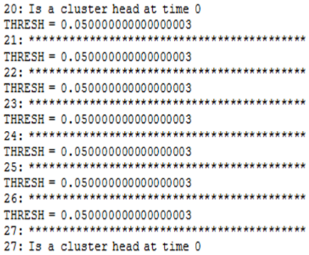
```
Node 0: (65.745973803916002,92.581722416254564)
Node 1: (21.008649990432268,92.380389195112684)
Node 2: (37.201202259027028,40.606367467253641)
Node 3: (71.218022131928251,61.297971318148988)
Node 4: (35.003944130150579,11.288995440718249)
Node 5: (34.146372151629237,98.076752432657756)
Node 6: (75.978135678906995,64.526355389750734)
Node 7: (94.45503554048716,5.7823289678349754)
Node 8: (83.602962402441989,14.98909784247591)
Node 9: (21.767438492629417,45.338745622587737)
Node 10: (8.2976788321033492,59.088131160982016)
Node 11: (94.220422624713009,62.643053551503947)
Node 12: (41.80104012684945,50.081411958710021)
Node 13: (18.290790039250062,13.30818967582108)
Node 14: (70.743881524886874,92.416788773805266)
Node 15: (48.968921345178465,20.661048414493468)
Node 16: (50.240702391714187,95.485097540302704)
Node 17: (18.034359867700545,3.486296443029445)
Node 18: (94.184317995879951,55.832556754272687)
Node 19: (77.781369061107455,71.469810032970187)
Node 20: (93.097224129874832,85.045950806255433)
Node 21: (67.295200735002382,30.438753185066748)
Node 22: (84.124781416787201,85.201271942444734)
Node 23: (77.777536668711122,7.0587910278974055)
Node 24: (37.10080587170124,53.244285682795699)
Node 25: (76.709470747369096,56.074851032381346)
Node 26: (50.021301233219582,8.0098267216281158)
Node 27: (21.157710403743067,97.638755709695985)
```

Fig. 9. Placement of nodes.



```
Node 98: (2.466682112993059,57.526273074339272)
Node 99: (44.071560420129238,10.71598111219517)
Max Distance for this Simulation is 139.0
```

Fig. 10. Total active time for the network.



```
20: Is a cluster head at time 0
THRESH = 0.050000000000000003
21: ******************************************
THRESH = 0.050000000000000003
22: ******************************************
THRESH = 0.050000000000000003
23: ******************************************
THRESH = 0.050000000000000003
24: ******************************************
THRESH = 0.050000000000000003
25: ******************************************
THRESH = 0.050000000000000003
26: ******************************************
THRESH = 0.050000000000000003
27: ******************************************
27: Is a cluster head at time 0
```

Fig. 11. Cluster heads and the energy threshold for each node.

VIII. Conclusion

1) The prediction model- may help in accurately predicting the advent of bursty traffic in a network at a given instance.
2) SMAC does perform better in terms of energy consumption when compared to the other mac protocol on a WSN.
3) LEACH we have run it on a network , we are yet to compare it to another mac protocol

References

[1] Georgios Z. Papadopoulos, Vasileios Kotsiou, Antoine Gallais, George Oikonomou,Periklis Chatzimisios, Theo Tryfonas and Thomas Noel ,A Mobility-Supporting MAC Scheme for Bursty Traffic in IoT and WSNs.

[2] Jing Jin ,Traffic Burst Prediction in Radio Access Network with Machine Learning, KTH Royal Institute Of Technology. School of Electrical Engineering

[3] Installation and usage steps for NS2, https://www.youtube.com/watch?v=7ghU6peU350

[4] Simulation of Wireless sensor. network in NS2, http://nipunharitash.com/technical/.

[5] Tutorials for Sensor MAC, https://www.youtube.com/watch?v=FmE1b7ETVd0 .

[6] Sensor MAC http://www.isi.edu/scadds/projects/smac/.

[7] LEACH protocol for Wireless sensor networks, www.radford.edu/ nsrl/creu0809/Presentations/LEACH.ppt.