Survey of Efficient MAC solutions for MANETs

A Project Report

submitted by

Ashish Katoch
(EE19M005)

in partial fulfilment of the
requirements for the award of
the degree of

MASTER OF TECHNOLOGY

DEPARTMENT OF ELECTRICAL ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY MADRAS

(June 2021)

THESIS CERTIFICATE

This is to certify that the thesis titled **Survey of Efficient MAC solutions for MANETs**, submitted by **Ashish Katoch**, to the Indian Institute of Technology, Madras, for the award of the degree of **Master of Technology**, is a bonafide record of the research work done by him under our supervision. The contents of this thesis, in full or in parts,have not been submitted to any other Institute or University for the award of any degreeor diploma.

Prof. K Giridhar
Project Guide
Professor
Dept. of Electrical Engineering
IIT-Madras, 600 036

Place: Chennai

Date: ..th June 2021

ACKNOWLEDGEMENTS

I would like to take this opportunity to thank our Parents & Navy for their support and encouragement without which learning in and becoming a part of such a prestigious institution would not have been possible. I would like to dedicate this work to them.

I am thankful to Dr. K Giridhar for his continual support and invaluable advice throughout the duration of our research. I feel honoured and encouraged to have worked under his guidance. Also we are very much thankful to Mr. Venkatesh for his help in the research.

ABSTRACT

MANET stands for Mobile adhoc Network also called as wireless adhoc network or adhoc wireless network that usually has a routable networking environment on top of a Link Layer ad hoc network. They consist of a set of static/mobile nodes connected wirelessly in a self-configured, self-healing network without having a fixed infrastructure. MANET nodes are free to move randomly as the network topology changes frequently. Each node behaves as a router as they forward traffic to other specified nodes in the network.

Designing a small scale and a large scale MANET requires efficient MAC protocol to define scheduling/decision rules at various nodes. Further, routing protocol is required to find least hop/optimal route from source node to destination node.

The report covers detailed survey of MAC protocols suitable for small scale MANET typically with 30-40 nodes in an area of 1 sq km as well as large scale MANET with 100-200 nodes in an area of 25 sq kms.

The report is further supported with simulation of large scale MANET in NS3 with 100 nodes in an area of 25 sq kms.

TABLE OF CONTENTS

CHAPTER 1

Introduction

A Mobile Ad Hoc Network (MANET) is an interconnected system of wireless nodes which communicate over bandwidth constrained wireless links. Each wireless node can function as a sender, a receiver or a router. When the node is a sender, it can send messages to any specified destination node through some route. As a receiver, it can receive messages from other nodes. When the node functions as a router, it can relay the packet to the destination or next router in the route. When necessary, each node can buffer packets awaiting transmission. The nodes move randomly, hence at a given point in time, an ad hoc network exists between the nodes, giving rise to an arbitrary network topology. MANETs can be dynamically formed among any group of wireless users and require no existing infrastructure or configuration.

A MANET has several marked characteristics. First, it does not have a centralized infrastructure. It is unlike the traditional mobile wireless networks in which base stations, access points and servers have to be deployed before the networks can be used. As shown in **Figure 1**, the ad hoc network is decentralized, with all mobile nodes functioning as routers and all wireless devices being interconnected to one another. Intuitively, this means that the MANET is also a self-configuring network in which network activities, including the discovery of the topology and delivery of messages, are executed by the nodes themselves.

Figure 1: An Adhoc network

The second characteristic of a MANET is that it has a dynamic topology. Nodes are free to move arbitrarily, causing the network topology to change rapidly and unpredictably over time. Alternative paths are automatically found, after which data packets are forwarded across the multi-hop paths of the network. MANETs use various routing mechanisms to accomplish this.

Thirdly, a MANET operates on bandwidth constrained variable-capacity links. Wireless links have significantly lower capacity than hard-wired links. As such, a MANET has relatively low bandwidth links, high bit error rates, and unstable and asymmetric links. This is in contrast to wired networks which are characterised by high bandwidth links, low bit error rates and stable and symmetric links. One effect of having a low link capacity is that congestion is typically the norm rather than the exception.

Fourthly, a MANET is often bound by energy constrained Operations. This is because its nodes are often hand-held battery-powered devices. Since the mobile nodes rely on these exhaustible means for energy, power conservation is important in a MANET system design.

Lastly, there is limited physical security. Mobile wireless networks are more prone to the physical security threats of eavesdropping, interception, denial-of service and routing attacks as compared to fixed-cable. Hence, security techniques have to be applied to reduce these threats. Nodes prefer to radiate as little power as necessary and transmit as infrequently as possible. This will decrease the probability of detection and interception. In addition, the decentralised nature of network control will add robustness against failure as opposed to the centralised networks.

Flow of thesis:

The thesis is organized as follows.

Chapter 2 discusses some basic classification of MAC protocols used for MANETs followed by survey of various multi-channel MAC protocols.

Chapter 3 discusses efficient MAC protocols suitable for small scale MANETs, typically of 30-40 nodes in 01 sq km area.

Chapter 4 discusses efficient MAC protocols suitable for large scale MANETs, typically of 100-200 nodes in 25 sq km area.

Chapter 5 discusses Market survey of some MANET SDRs available in India & abroad.

Chapter 6 gives results of simulation of large scale MANET in NS3 with 100 nodes in an area of 25 sq Kms

Chapter 7 gives conclusion & future scope

CHAPTER 2

CLASSIFICATION OF MANET MAC PROTOCOLS

This chapter briefly discusses some of the basic classification of MAC protocols used for basic MANETs followed by survey of multi-channel MAC protocols.

2.1 Basic Classification of MAC Protocols

MAC layer, sometimes also referred to as a sub-layer of the 'Data Link' layer, involves the functions and procedures necessary to transfer data between two or more nodes of the network. It is the responsibility of the MAC layer to perform error correction for anomalies occurring in the physical layer. The layer performs specific activities for framing, physical addressing, and flow and error controls. It is responsible for resolving conflicts among different nodes for channel access. Since the MAC layer has a direct bearing on how reliably and efficiently data can be transmitted between two nodes along the routing path in the network, it affects the Quality of Service (QoS) of the network. The design of a MAC protocol has to address issues caused by mobility of nodes and an unreliable time varying channel.

Various MAC schemes developed for wireless ad hoc networks can be classified as shown in **Figure 2**. In contention-free schemes (e.g., TDMA, FDMA, CDMA), certain assignments are used to avoid contentions. Contention based schemes, on the other hand, are aware of the risk of collisions of transmitted data.

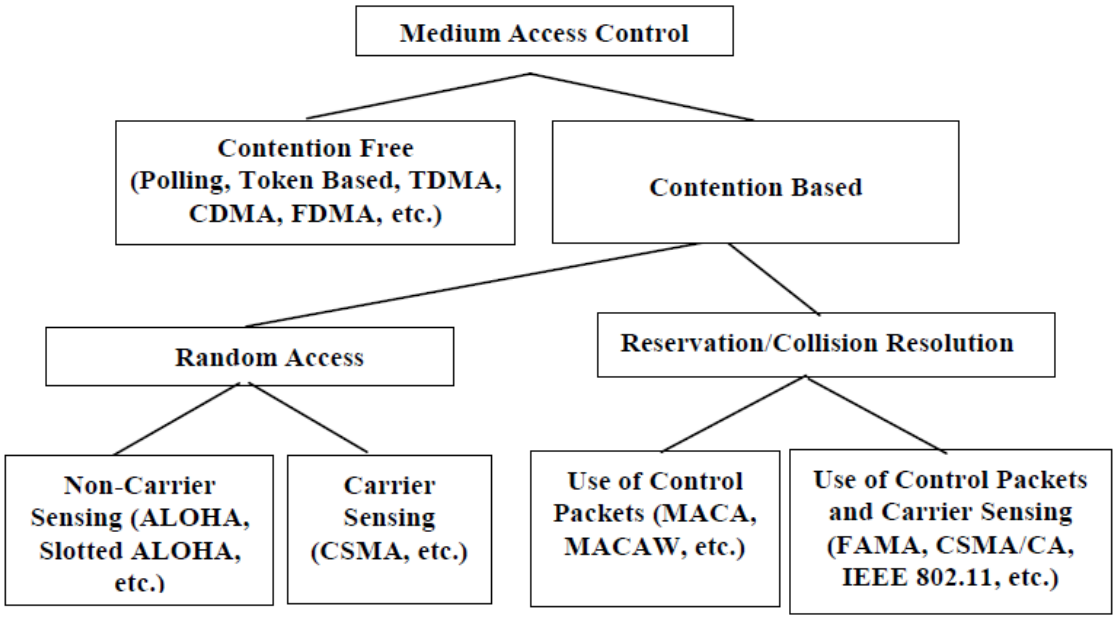


Figure 2: Basic MAC Classification

For accessing the shared wireless medium in ad hoc networks, two families of medium access control (MAC) protocols are dominant. The first family is contention-based protocols, typically using Carrier Sense Multiple Access (CSMA) technique. Such MAC protocols use available bandwidth on demand and are very flexible and efficient for low traffic load conditions and small network sizes. When network size increases and network traffic is high, CSMA-based protocols are not able to satisfy QoS requirements, implying that CSMA-based protocols are not scalable.

A second family of MAC protocols for ad hoc networks is contention-free protocols, usually based on the Time-Division Multiple Access (TDMA) mechanism. TDMA-based medium access is one of the most common medium access methods where the wireless medium is time-shared by all nodes. Channel bandwidth in the network is divided into time frames, called superframes, with every superframe further partitioned into time slots. Multi-frequency TDMA (Mf-TDMA) extends the basic TDMA medium access method, which uses only one frequency channel, to multiple channels. Slots in a Mf-TDMA superframe are represented as time-frequency tuples. In TDMA-based protocols each node transmits only during slots allocated to it, avoiding any contention for accessing the shared medium.

Compared to CSMA-based protocols, TDMA-based protocols mitigate internal collisions and thus improve delivered QoS for large-scale networks with high traffic demands. Due to its favorable properties in terms of scalability, TDMA scheduling techniques have gained attention for larger ad hoc networks in recent years. However, the reliability and throughput of networks with TDMA access schemes may still be impacted by external interference or the occurrence of exposed/hidden nodes. For the function of slot allocation in TDMA schemes, there are static and dynamic algorithms. As ad hoc networks need to support constant changes in traffic demands and network topology, dynamic scheduling algorithms are known to outperform static scheduling algorithms. There exist two main models for handling dynamic TDMA scheduling: centralized and distributed. Centralized models consist of one or more control nodes that gather information about the network state and make scheduling decisions that are advertised to each node. In distributed models, decision making is done at the node level based on local information on the network without requiring any centralized control; nodes exchange information about slot usage with their neighbors in order to take distributed decisions on slot allocation.

Even though centralized scheduling protocols can offer close to optimal solutions for some use cases as they have global knowledge of network topology and traffic patterns, they are not suitable for networks with frequently changing topology and traffic demands over time. Changes in network topology or traffic patterns result in continuous schedule recalculations and increased control overhead, thus leading to degraded network performance. Moreover, centralized scheduling protocols are not scalable as they incur high control overhead for large-scale wireless networks. In dynamic and large ad hoc networks, distributed slot allocation algorithms are preferred to cope with scalability and changes in the network topology. Also, distributed algorithms are more fault-tolerant, as a major problem in centralized algorithms is the existence of a single point of failure; if the central control node fails or disconnects, slot scheduling cannot be executed anymore. In any case, whereas many distributed scheduling protocols are proposed so far, an increase in size and/or density of wireless networks still induces scalability issues for existing protocols. The most common reason for the scalability issues in large-scale ad hoc networks is their multi-hop nature,

which highly depends on network size and packet forwarding capabilities. Other various classes of MAC protocols are listed in **figure 3** below.

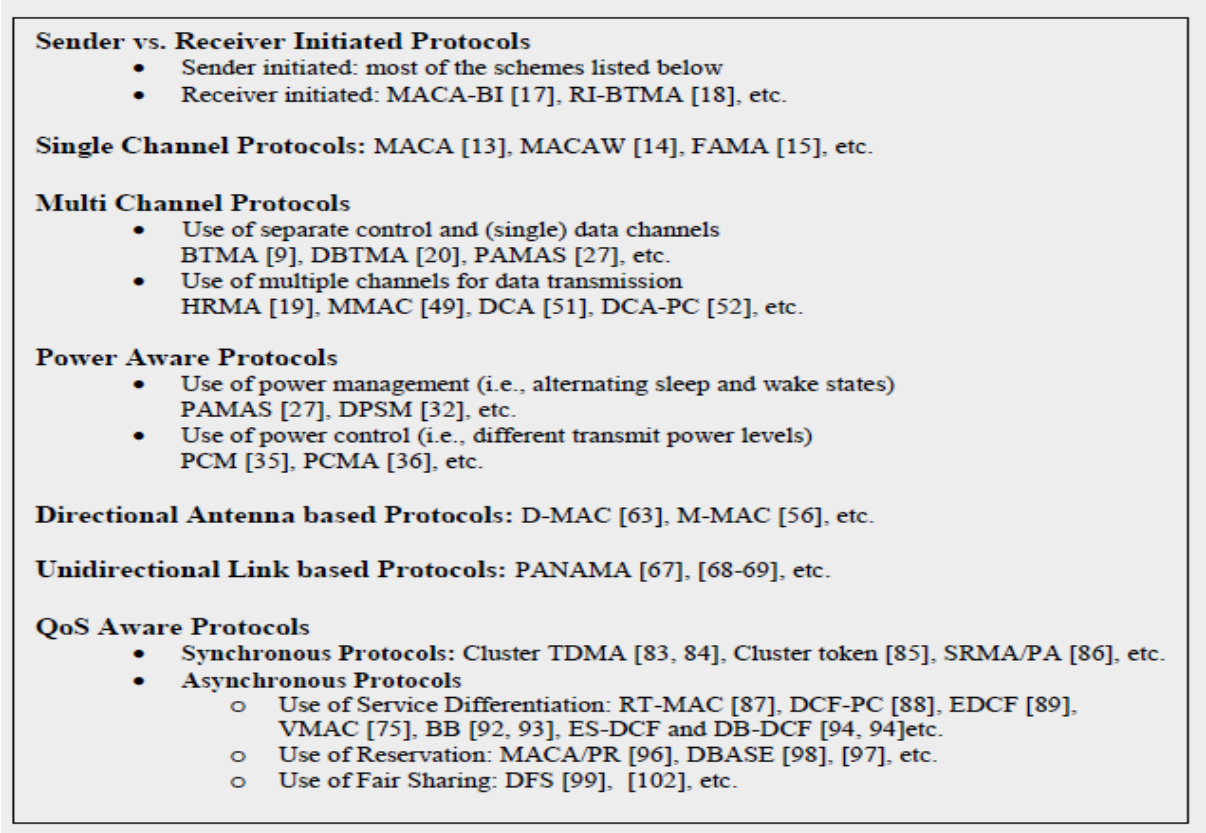


Figure 3: Other Classes of MAC protocols

The above mentioned protocols given in figure 2 & 3 were studied and it was found instead of single channel MAC protocols, the multichannel MAC protocols offer several advantages as listed below:-

(a) Firstly, multiple channels could simply be a way to make use of additional bandwidth, thus increasing the global throughput of a network. For example, the IEEE 802.11b physical layer has 14 channels, 3 of which are orthogonal and thus available for concurrent use. Instead of statically assigning different frequencies to separate BSSs, one could use the additional channels to create a single BSS of higher capacity. Thus the allocation of the channels would be done dynamically and in a distributed way

(b) A second motivation for the use of multiple channels is that they could provide some performance improvements with respect to a single-channel CSMA & TDMA even in the case of fixed aggregated bandwidth. The idea is that more channels, by allowing concurrent transmissions, could reduce the number of collisions, and bring about a more efficient utilization of the bandwidth. Moreover, the hidden terminal problem, which seriously affects the performance of wireless LANs, could be relieved by an appropriate allocation procedure.

2.2 Survey of Multi-channel MAC protocols

Multi-channel wireless networks operate by partitioning the available spectrum into many channels of equal bandwidth. The bandwidth of different channels can be made different also to make it suitable to a particular type of network or ad-hoc network. The Multiple-Channel MAC layer rules and associated protocols can be divided into various categories depending upon the mechanism used to implement them. The methods used to categorized them depends upon the number of transceivers used (single or multiple transceivers), Type of rendezvous, communication initiation (sender or receiver), Type of control Channel used and Type of synchronization method used. Below is a brief discussion of all these methods.

2.2.1 Dedicated Control Channel

In dedicated control channel MAC protocols each device uses at least two or more communication channels. One among them is for control and others are for data communications. In principle, all devices can over- hear all the agreements made by other devices, even during data exchange. This system's efficiency is limited only by the contention for the control channel and the number of available data channels. **Figure 4** illustrates the operations of Dedicated Control Channel. In the figure, channel 0 is the control channel and channels 1, 2, and 3 are for data transmission. When device A wants to send to device B, it transmits an RTS (request-to-send) packet on the control channel. That RTS specifies the lowest-numbered free channel. Upon receiving the RTS, B responds with a CTS (clear-to-send) packet on the control channel, confirming the data channel suggested by A. The RTS and CTS packets also contain a Network Allocation Vector (NAV) field, as in 802.11, to inform other devices of the duration for which the sender, the receiver, and the chosen data channel are busy. Since all devices listen to the control channel at all times, they can keep track of the busy status of other devices and channels even during data exchange. Devices avoid busy channels when selecting a data channel.

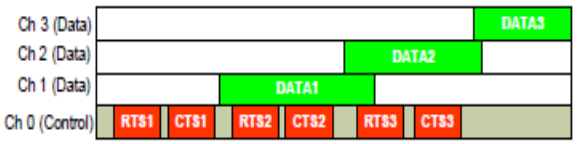


Figure 4: Dedicated Control Channel MAC protocol

2.2.2 Split Phase Protocols

In this approach, devices use a single radio. Time is divided into an alternating sequence of control and data exchange phases, as shown in **Figure 5**. During a control phase, all devices tune to the control channel and attempt to make agreements for channels to be used during the fol- lowing data exchange phase. If device A has some data to send to device B, it sends a packet to B on the control channel with the ID of the lowest numbered idle channel, say, i. Device B then returns a confirmation packet to A. At this point, A and B have agreed to use channel i in the upcoming data phase. Once committed, a device cannot accept other agreements that conflict with earlier agreements. In the second phase, devices tune to the agreed channel and transfer data. The protocol allows multiple pairs to choose the same channel because each pair might not have enough data to use up the entire data phase. As a result, the different pairs must either schedule themselves or contend during the data phase. In the analysis, we assume that at most one device pair can be assigned to each channel, so there

is no need for scheduling or contention. The advantage of this approach is that it requires only one radio per device. However, it requires time synchronization among all devices, though the synchronization can be looser than in Common Hopping because devices hop less frequently.

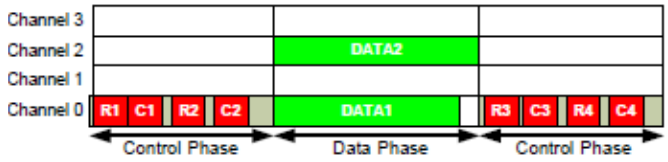


Figure 5: Split-Phase MAC protocol

2.2.3 Common Hopping Protocols

In this approach, devices have only one radio. Devices not exchanging data cycle through all channels synchronously. Pair of devices stop hopping as soon as they make an agreement for transmission and rejoin the common hopping pattern subsequently after transmission ends. The Common Hopping protocol improves on Dedicated Control Channel in two respects: 1) it use all the channels for data exchange; 2) it requires only one transceiver per device. As shown in **Figure 6**, the hopping pattern cycles through channels 0, 1, 2 and 3. When device A wants to send to device B, it sends an RTS to B on the current common channel. If B receives the RTS properly, it returns a CTS on the same channel. Devices A and B then pause hopping and remain on the same channel during data transfer while the other idle devices continue hopping. When they are finished, devices A and B rejoin the common hopping sequence with all the other idle devices. It is possible that the common hopping sequence wraps around and visits the channel A and B are using before they finish data exchange. Idle devices sense the carrier and refrain from transmitting if it is busy. While A and B are exchanging data, they are unaware of the busy status of the other devices. Hence, it is possible that a sender sends an RTS to a device that is currently busy on a different channel. Another issue with this approach is that devices hop more frequently. State-of-the-art integrated circuits implementations of tri-mode 802.11a/b/g radios require only about 30μsec for its voltage-controlled oscillator (VCO) to settle, but commercial off-the-shelf 802.11b transceivers require about 150 to 200μsec to switch channels. Considering that an RTS in 802.11b takes only about 200 − 300μsec, the hopping time penalty is not negligible. The approach also requires devices to have tight synchronization.

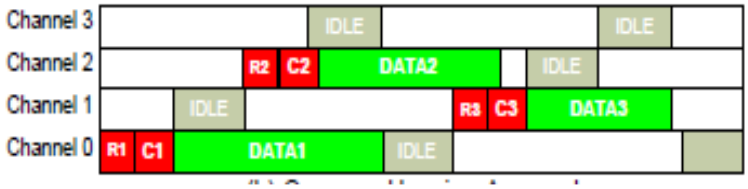


Figure 6: Common Hopping approach

2.3 Comparison of Multi-channel MAC protocols

The comparison of all three types is summarized in table given in **figure 7**. Further, basic protocol from all the three categories were simulated in 802.11b scenario with 20 nodes and 03 channels, each having data rate equal to 2 Mbps. The **figure 8** shows aggregate

throughput of all the three classes of protocols calculated analytically and through simulation.

Parameter Investigated	Dedicated Control Channel	Split Phase	Common Hopping
Ability to Use Many Channels	Good for long pkts Limited for short pkts	Limited	Limited
Sensitivity to Packet Len (Lower is better)	High if many avail. channels Low if few avail. channels	High	High
Sensitivity to Channel Switching Time (Lower is better)	Low	Low	Very High
Sensitivity to Receiver Contention (Lower is better)	Low	Medium	High

Figure 7: Comparison of three categories of Multi-channel MAC protocols

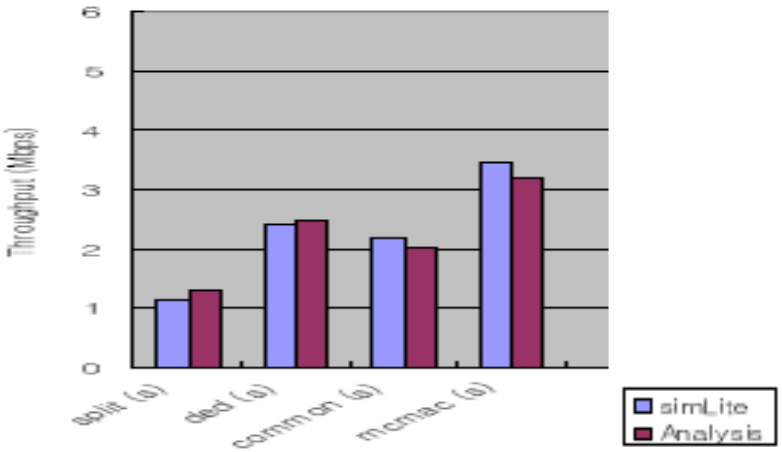


Figure 8: Aggregate throughput of all three approaches

Based on the above comparison, the MAC protocols for small scale MANETs are selected from these categories which give optimal solution are explained in next chapter.

CHAPTER 3

EFFICIENT MAC PROTOCOLS FOR SMALL SCALE MANETS

Based on literature survey & outcomes from chapter 2, the following MAC protocols are found to be suitable for supporting a small scale MANET. These protocols are based on the categories explained in chapter 2.

3.1 CAM-MAC

Protocol Description. CAM-MAC is Cooperative Asynchronous Multichannel MAC protocol designed by Temasek Defence Science Institute (National University of Singapore) and Bell Labs Research (Bengaluru) in 2009. The protocol has been also implemented on COTS test bed/hardware in order to confirm its viability. CAM-MAC uses a new concept of Distributed Information Sharing (DISH), which is a distributed flavor of control-plane cooperation, as a new approach to wireless protocol design, and then apply it to multichannel medium access control (MAC) to solve the MCC (multi-channel conflict) problem. The basic idea is to allow nodes to share control information with each other such that nodes can make more informed decisions in communication. This notion of control-plane cooperation augments the conventional understanding of cooperation, which sits at the data plane as a mechanism for intermediate nodes to help relay data for source-destination pairs. Applying DISH to multichannel ad hoc networks, neighboring nodes who identify an MCC problem tend to notify the transmitter-receiver pair of the problem to avoid collisions and retransmissions. **Fig. 9** below gives an illustration. Two node pairs, (U1, U2) and (V1, V2) are performing data exchanges on channels 1 and 3, respectively, and node A1 is to initiate a communication with A2 at this moment. If A2 is on a channel different from A1, a deaf terminal problem is created. If (A1, A2) selects channel 1 or 3 for data exchange, a channel conflict problem is created. In either case, the neighboring nodes C, D, or E may have relevant channel usage information and could share with (A1,A2) to solve the MCC problem.

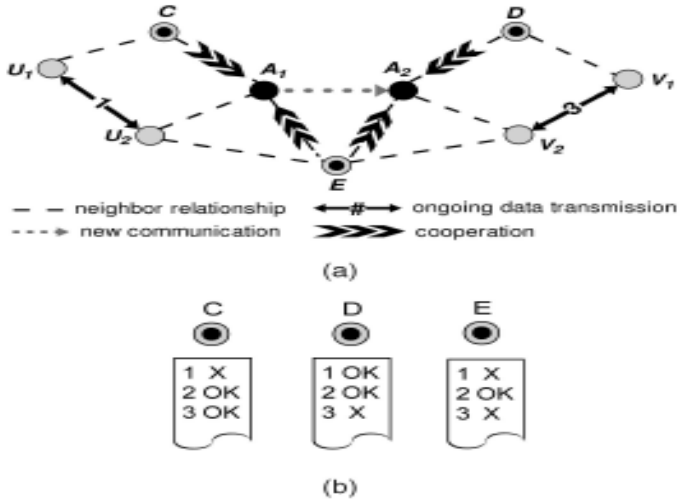


Fig. 1. An illustration of the DISH idea. (a) A multichannel scenario. (b) Knowledge at individual nodes. By consolidating the knowledge at nodes C and D, or acquiring knowledge from node E, it shows that the conflict-free channel is channel 2.

Figure 9: Cooperative decision making

The following assumptions have been taken into consideration while designing the protocol:-

(a) Each node is equipped with a single half duplex radio that can dynamically switch between a set of orthogonal frequency channels but can only use one at a time. CAM-MAC usually works on 3 Channels (1-Dedicated Control Channel and 2- Data Channels) for multihop scenario.
(b) Two channel selection strategies i.e RAND, where a node randomly selects one from a list of channels that it deems free based on its knowledge and MRU (most recently used), where a node always selects its MRU data channel unless it finds the channel to be occupied by other nodes, in which case RAND selection strategy is used.
(c) They have not assumed any (regular) radio propagation patterns, nor assumed any relationship between communication ranges and interference ranges. Intuitively, none of the nodes is responsible for providing cooperation; a node cooperates if it can (if it is idle and overhears a handshake that creates an MCC problem), and simply does not cooperate otherwise. Actually, there often exists at least one neighboring node that can cooperate, and even in the worse where no one can cooperate, the protocol still proceeds (as a traditional non-cooperative protocol).

The protocol design has one channel which is designated as the control channel and the other channels are designated as data channels. A transmitter and a receiver perform a handshake on the control channel to set up communication and then switch to their chosen data channel to perform a DATA/ACK handshake, after which they switch back to the control channel. The control channel handshake is depicted in **Fig. 10**.

A transmitter sends a PRA and its receiver responds with a PRB, like IEEE 802.11 RTS/CTS for channel reservation. Meanwhile, this PRA/PRB also probes the neighborhood inquiring whether an MCC problem is created (in the case of a deaf terminal problem, it is probed by PRA only). Upon the reception of the PRA or PRB, each neighbor performs a check and, if identifying an MCC problem, sends an INV message to invalidate the handshake (the receiver can also send INV after receiving PRA, since it is also one of the transmitter's neighbors).

If no INV is sent and the transmitter correctly receives PRB, it sends a CFA to confirm the validity of PRA to all its neighbors (including the receiver), and the receiver will send a CFB to confirm the validity of the PRB if it correctly receives CFA. This marks the end of a control channel handshake. If any INV is sent, the handshake will not proceed and the transmitter will back off. The NCF is merely used by the transmitter to inform its neighbors that the PRA and CFA are invalid when it fails to receive CFB (the receiver gets INV after sending PRB).

The cooperative collision avoidance period is for mitigating INV collision caused by multiple neighbors sending INVs simultaneously. It is a simple CSMA-based mechanism where each neighbor schedules to send INV at a random point in this period and continues sensing the channel. Once the node that schedules at the earliest time starts to send, others in its vicinity cancel sending their INVs (a receiver can also cancel its PRB).

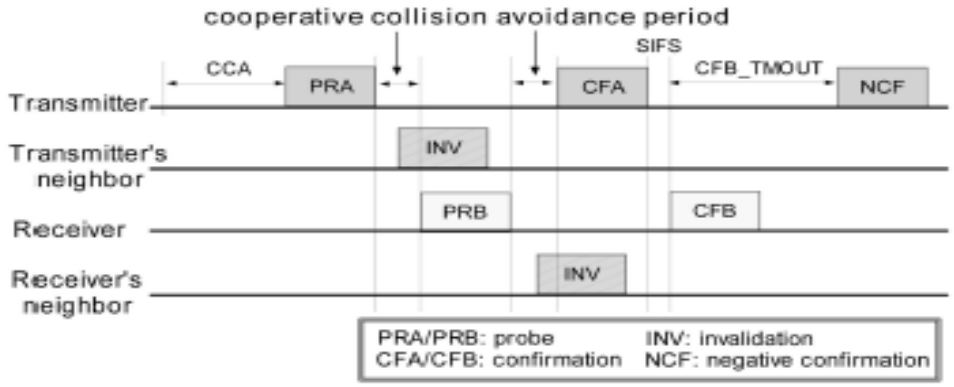


Figure 10: CAM-MAC control channel handshake

A possible set of frame formats is shown in **Fig. 11**. Both PRA + CFA and PRB + CFB carry the channel usage information of a communication being established, and an INV carries the channel usage information of an established communication that is to be collided (in the case of a channel conflict problem) or engages the receiver (in the case of a deaf terminal problem). A node may overhear this channel usage information and will cache it in the node's channel usage table, shown in **Fig. 12**. Note that until column does not imply clock synchronization. It is calculated by adding the duration in a received CFA/ CFB/INV message to the node's own clock. Similarly, when sending INV, a node does a reverse conversion from until to duration using a subtraction. Also note that this table is by caching overheard information while not by sensing data channels. This is because sensing data channels often obtains different channel status at the transmitter and the receiver, and resolving this discrepancy adds protocol complexity. In addition, this may lead to more channel switching and radio mode (TX/RX/IDLE) changes and thus incurs longer delay.

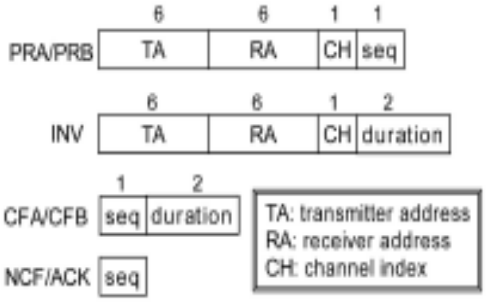


Figure 11: Set of frame formats

TA	RA	CH	until
A_1	A_2	1	11:30:52
B_1	B_2	3	11:30:56

Figure 12: Channel usage table

Simulation results. CAM-MAC has been analyzed for both sigle hop and multi-hop networks. However, we are considering only multi-hop scenario to our problem statement. For multihop environment the performance comparison has been done between five protocols first namely IEEE 802.11, CAMMAC-RAND, CAMMAC-MRU, UNCOOP-RAND, UNCOOP-MRU, using a discrete event simulator which was developed on Fedora Core 5 with a Linux kernel of version 2.6.9. The protocol UNCOOP is identical to CAM-MAC except that the cooperation element is removed. CAM-MAC usually is designed for 3 Channels (one control and two data channels). However, channels are varied in simulation just to check when the protocol attains saturation. Three performace metrics have been siumlated namely, aggregate (end-to-end) throughput, data channel conflict rate defined as the packet collisions on data channels per second over all nodes, packet delivery ratio which is defined as the number of data packets successfully received by destinations normalized by the no. of data packet sent by sources. The simulation parameters are given below:-

(i) Nodes – 360 with node density $10/r^2$, where r is the transmission range.
(ii) Area – 1.5 * 1.5 Kms
(iii) Node tx range – 250 m
(iv) Node interference range – 500 m
(v) N nodes form N disjoint flows randomly
(vi) Routing- Shortest Path Multihop
(vii) 01 control channel and 05 data channels (however it operates with total 03 channels only)
(viii) Bandwidth/channel- 1 Mbps (each)
(ix) Each source generates 2 kbyte payload according to poisson point process.
(x) Collision avoidance period – 35 us
(xi) Rest parameters such as PLCP, SIFS limit etc as per IEEE 802.11 MAC
(xii) Each simulation is terminated when a total of 100,000 data packets are sent over network and all results are averaged over 15 randomly generated networks.

(a) Effect of Traffic load:-

The traffic rate is varied from 2.5 to 50 kbps per flow. The throughput, data channel conflict rate and packet delivery ratio compared with traffic generation rate per flow is shown in figure below:-

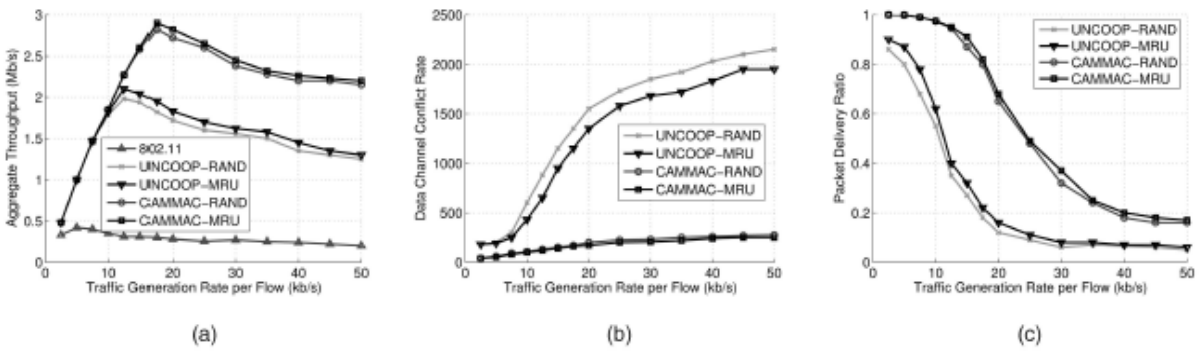


Figure 13: Effect of traffic rate on various parameters

(b) Impact of data payload size:-

There are 360 nodes and the traffic load is 20 kbps. The payload size is varied from 256 to 8192 bytes. The throughput, data channel conflict rate and packet delivery ratio compared with payload size flow is shown in figure below:-

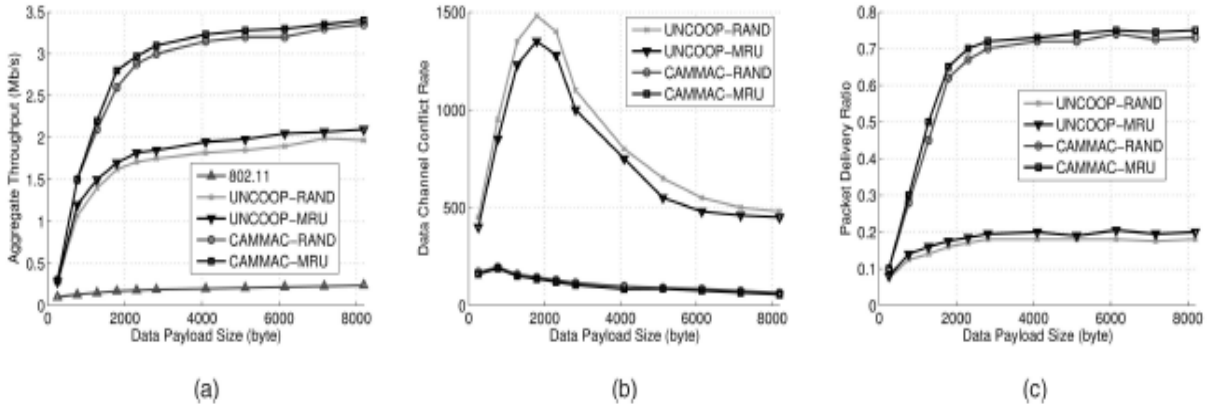


Figure 14: Impact of data payload on various parameters

(c) Impact of Node Density:-

Node density is varied from 2 to $20/r^2$ and fixed traffic load 20 kbps. The throughput, data channel conflict rate and packet delivery ratio compared with node density is shown in figure below:-

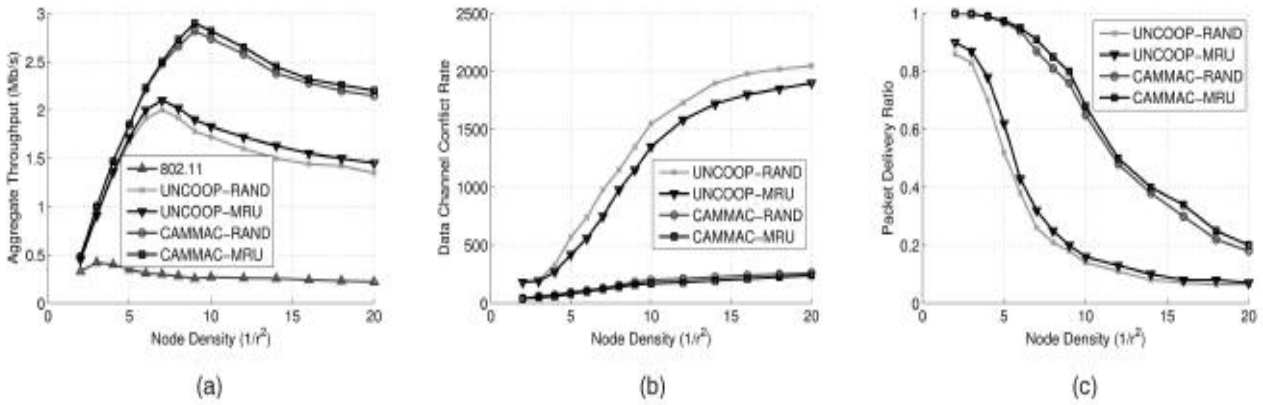


Figure 15: Impact of node density on various parameters

CAM-MAC effectively mitigates MCC problems and substantially enhances system performance.

Comparison with MMAC, SSCH and AMCP protocols (category- split phase, common hopping and dedicated channel protocols respectively). MMAC and SSCH require clock synchronization while AMCP does not. All protocols are using single half duplex trans-receiver.

(a) Comparison with MMAC (Multi-channel MAC). The simulation is carried out with 100 nodes in area of 500*500m (multi-hop scenario), where 40 sources and 40 destinations are randomly chosen. CAM-MAC achieves 1.57 times higher throughput then MMAC. Number of channels are 4 and packet size is 1024 bytes.

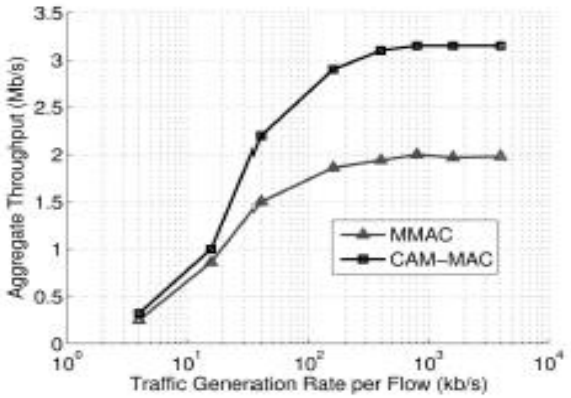


Figure 16: CAM-MAC vs MMAC

(b) Comparison with common hopping protocol SSCH (Slotted seed common hopping). Since SSCH uses 13 channels hence here CAM-MAC is compared with SSCH with parameters namely, channel capacity (54 Mbps), packet size (512 bytes) and channel switching delay (80 us). The results show that CAM-MAC outperforms SSCH by a factor of 1.5.

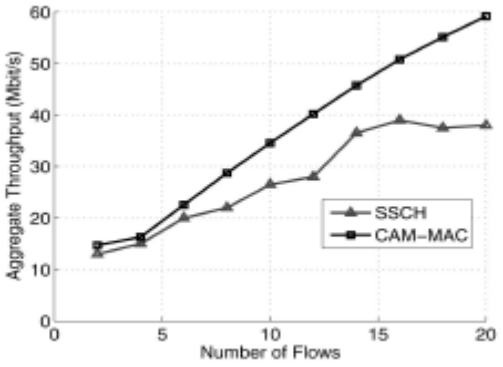


Figure 17: CAM-MAC v/s SSCH

(c) Comparison with AMCP (dedicated control channel category). For comparison with AMCP (Asynchronous Multi-Channel Coordination Protocol), 30 nodes forming 15 non-disjoint flows in a single hop network scenario is considered. The number of channels are varied from 2 to 12 in order to check saturation limit. The channel capacity of each channel is 2 Mbps, packet size is 1000 bytes and channel switching delay is 224 us. CAM-MAC saturates at 5 Mbps as compared to AMCP at 4.2 Mbps as shown in figure below.

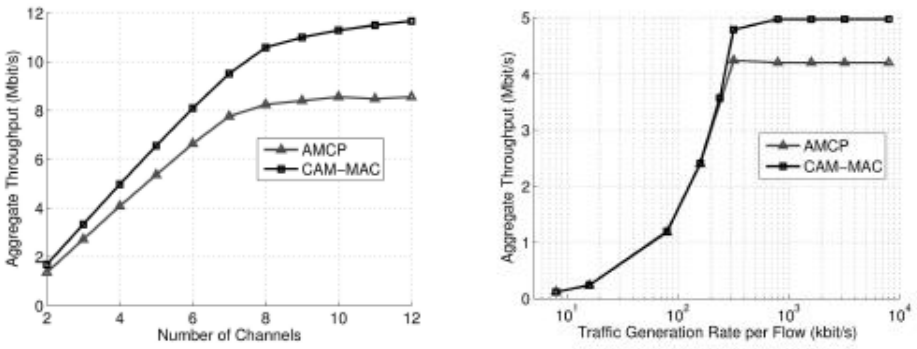


Figure 18: CAM-MAC vs AMCP

Hardware Implementation. A microcontroller based platform Telos B mote with an ASIC radio CC2420 as hardware platform is used. Tiny OS 2.0 is used as software. TelosB mote is IEEE 802.15.4 compliant RF trans-receiver (2.4 to 2.4835 GHz) supporting 250 kbps data rate. It has 8 MHz microcontroller with 10Kb RAM and 1 Mb external flash. Tiny OS is a small, open-source, energy-efficient software operating system developed by UC Berkeley which supports large scale, self-configuring sensor networks. Tiny OS 2.0 has almost full control over the MAC layer and its component architecture and C like programming enables rapid development.

Figure 19: Telos B Mote

(a) There are two limitations of the hardware. First, the maximum packet size that CC2420 supports is only 127 bytes. To overcome this, authors transmit a sequence of fragments as the substitution of a long data packet. The interval 't' between the fragments are counted as actual payload via 'tC', where C = 250 Kbps is the channel bandwidth, and the intermediate fragments are counted as pure payload without frame headers and footers. The second limitation is that the accuracy of timing on TelosB motes is not reliable at the microsecond level while reliable at the millisecond level. Authors circumvent this by proportionally prolonging all intervals, such as SIFS, CCA, and fragment intervals, up to milliseconds. Consequently, to transmit a 2-Kbyte data packet, a node transmits a sequence of 20 fragments with the length of 30 bytes each (including preamble) and the 19 intervals of 8ms each. This results in a total of 175 ms to transmit a data packet (each fragment needs 100-200 us to be sent in the air after assembled in memory). Under the same setting, a control channel handshake lasts 9 ms. The ratio between these two durations is close to that in the simulations. The collision detection technique is interleaved fragment sequence detection.

(b) The key idea is based on the fragmented data transmission and the large difference between the fragment interval (8 ms) and the fragment transmission time (<

1 ms), as described in above serial. As such, if a node receives a sequence of fragments from more than one transmitter, as illustrated by Fig. 20, it indicates a data packet collision (since intervals are actual payload). Therefore, packet collision can be easily detected by simply
checking fragment headers.

Fig 20. Packet collision detection via an interleaved fragment sequence, where TX/RX IDs are alternate and seq's are inconsecutive

(c) For visualization purposes, three LEDs are used on each TelosB mote to indicate specific events of interest (a maximum of $2^3 = 8$ events can be represented). For example, a blue LED indicates an ongoing control channel handshake, a green LED indicates an ongoing data channel handshake, and a red LED indicates transmitting a cooperative message. Other events are indicated by LED combinations. Fig. 21 is a snapshot in an experiment. In the experiments, the transmission power is 0 dBm which is the maximum on CC2420. Nodes are configured as disjoint flows in an indoor area, and source nodes are always backlogged. Three channels are used as one control channel and two data channels, each with bandwidth 250 Kbps. In collecting statistics for each of the four protocols, each single data point is by averaging over six experiments and each experiment runs for 360 actual seconds. The experimental results are presented in Fig. 22. When the number of nodes is four, the two MRU protocols have about twice throughput of the two RAND protocols. This is because MRU strategy in effect assigns each pair a dedicated data channel, while RAND strategy encounters channel conflicts with probability 0.5 at each selection (there are two data channels). The reason why CAMMAC-RAND and UNCOOP-RAND perform the same is that, any time when a transmitter-receiver pair selects a channel conflicting with the other pair, there is no additional node on the control channel to cooperate.
CAM-MAC achieves higher throughput as the number of nodes are increased as cooperation increases.

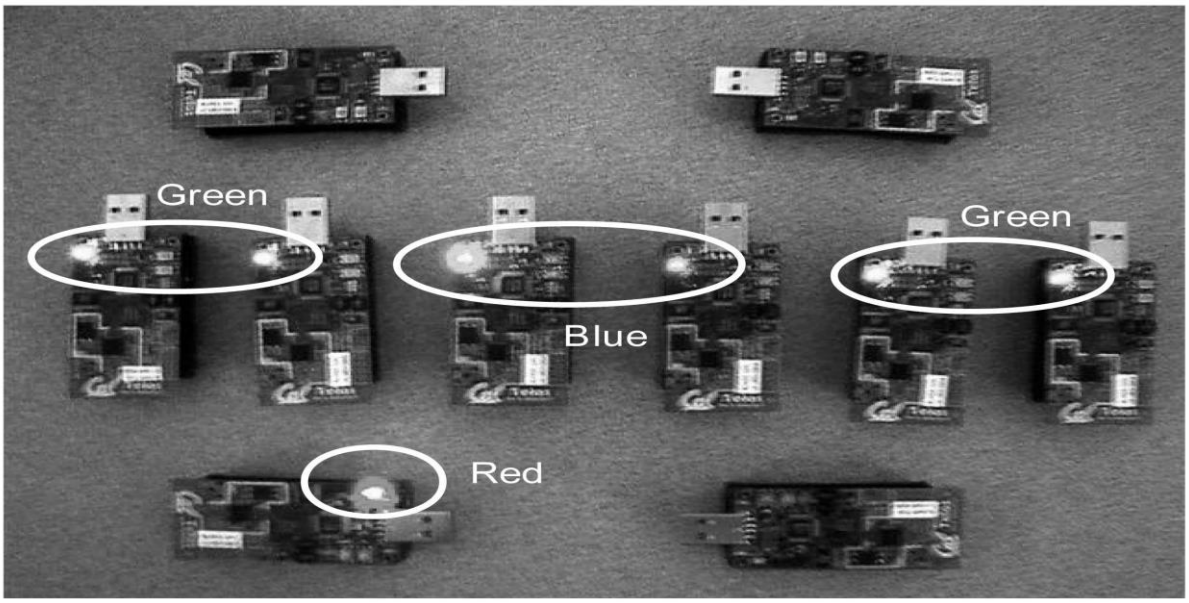


Fig. 21 above: A snapshot in an experiment on CAM-MAC with 10 nodes. The four "green nodes" are two transmitter-receiver pairs communicating on two different data channels. The two "blue nodes" are performing a control channel handshake (specifically, a PRA was just sent from one to the other). This creates a channel conflict problem since there are only two data channels which are already being in use. At this moment, a neighboring node, indicated by the red LED, identifies this (via the PRA) and sends a cooperative message (INV). Then, the two blue nodes will back off to discontinue the control channel handshake, and thus, data collision is prevented.

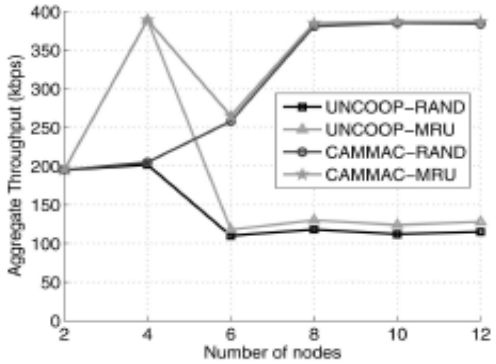


Fig. 22: Experimental results. The maximum utilizable bandwidth is 500 Kbps

(d) Impact of Mobility. One simple way of adapting CAMMAC to a mobile environment is to accordingly increase the frequency of updating neighbor information. Multihop simulations using random waypoint model with the same setup were conducted. Each node moves at a speed uniformly distributed in (0, 10] m/s and toward a randomly chosen target point for each movement. Each node independently updates neighbor information every 8 seconds. The results showed only a marginal (3 percent - 8 percent) performance degradation in comparison to the static scenario.

(e) Energy consumption. CAM-MAC has energy saving mode where nodes stay in sleep modes when not in operation/use.

3.2 DARMAC

Protocol Description. DARMAC is Distributed Asynchronous Reservation MAC research work undertaken by Tsinghua-QualComm joint research group at Tsinghua University (China) in 2011.

In DARMAC the nodes share their information about network space in a dedicated control channel and cooperatively select a collision free channel for the wants to transmit. In a DARMAC enabled network, each node is equipped with a single half-duplex transceiver that can dynamically switch between a set of orthogonal frequency channels but can only use one at a time. DARMAC is a single rendezvous protocol, which means that one and only one channel is designated as the control channel and the other channels are designated as data channels. As shown in Fig. 23, a transmitter and a receiver perform a handshake on the control channel to set up communication and then switch to their chosen data channel. The first data is transmitted by performing CSMA/CA at data channel because there may be ongoing data transmissions. Besides the first data packet, all data packets are transmitted directly without CSMA/CA. Data packets are protected by a DATA/ACK handshake. After data transmissions, the transmitter and receiver pair switches back to the control channel. In

DARMAC, a data channel transmission opportunity can be reserved, which means that a transmitter and receiver pair can perform control channel handshake to reserve a transmission opportunity on a data channel that is currently busy.

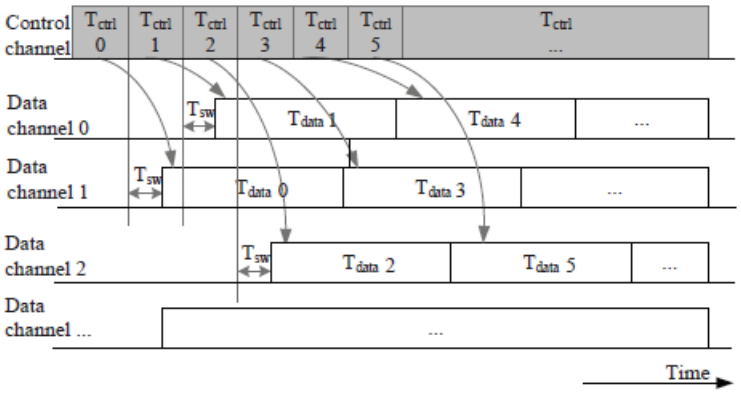


Fig23: Architecture of DARMAC

(a) Control Channel Handshake. The control channel handshake is depicted in Fig. 24.

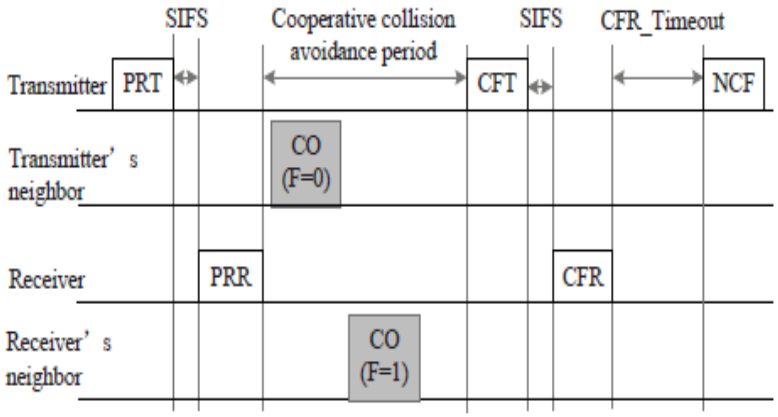


Fig24. DARMAC control channel handshake

A transmitter sends a probe named by PRT and its receiver responds immediately with a PRR. Due to transceiver switch, a small delay is incurred between a receiver receives PRT and starts PRR transmission, the delay is called SIFS, which is borrowed from IEEE 802.11 series of standards. Following PRR is a cooperative collision avoidance period, during which the transmitter's neighbor and receiver's neighbor check if there is a MCC problem. If identifying an MCC problem, a transmitter's neighbor or receiver's neighbor will send a CO message to notify its knowledge about the channel selected in the PRT/PRR handshake. (The receiver can also send CO after receiving PRT, since it is also one of the transmitter's neighbors). The CO message sent by a receiver's neighbor is also marked with a forwarding flag. The receiver always forwards the first CO with the forwarding flag true to the transmitter. After receiving a CO, the transmitter re-assures whether the handshake will continue or not. If the selected channel is currently reserved by other transmitter and receiver pairs and the reservation time is beyond a given threshold, it will reselect a channel and restarts the control handshake. If the transmitter correctly receives PRR and cooperative collision avoidance period timeout, it sends a CFT to reserve a channel time to all its neighbors (including the receiver), and the receiver will send a CFR to confirm the validity of the CFT and reserve a channel time to all its neighbors. This marks the end of a control

channel handshake. If the transmitter does not receive PRR, the transmitter will restart the handshake after PRR timeout. If the transmitter does not receive CFR after CFR timeout, the handshake will not proceed, in this case, the transmitter will transmit a NCF to inform its neighbors that CFT is invalid.
Before a control handshake, the transmitter needs to select a data channel. There are many channel selection strategies available. One simple yet effective is random selection, where a node randomly selects one from a list of channels that it deems free based on its knowledge. If no channel is free, the least occupied channel will be chosen. Random channel selection is used in the simulations for performance evaluation.
Before transmitting PRT, CFT and NCF, the transmitter performs CSMA/CA to reduce control packet collision. The neighbors of the transmitter and receiver will also perform CSMA/CA before transmitting CO. However, the receiver performs no CSMA/CA before transmitting CFR, CO.

(b) Frame formats. The frame formats are shown in Fig 25.

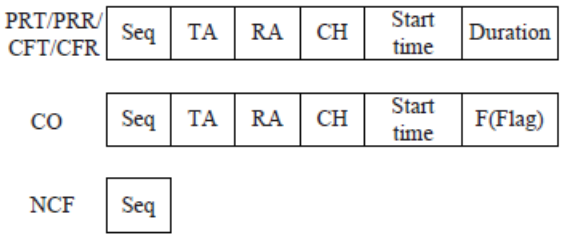


Fig25. DARMAC control channel handshake Packets

The field packets are given below:-

- Seq. It is a sequence number that identifies a control channel handshake initialized by a given transmitter.
- TA. It is the transmitter address.
- RA. It is the receiver address.
- CH. It is the selected data channel index of the ongoing handshake.
- Start Time. It is the start time offset of the transmission opportunity being applied.
- Duration. It is the duration of transmission opportunity being applied.
- F(Flag). It is the forwarding flag. A receiver's neighbor transmits a CO packet with F=1 when it cannot hear from the transmitter.

The quintuple *{TA, RA, CH, Start Time, Duration}* uniquely identifies a transmission opportunity being applied in a control handshake. Every node will maintain a list of quintuples. The list is being updated whenever a CFT, CFR, NCF, CO is received. A node will send CO when it receives a PRT or PRR and find parts of the quintuple the received packet carries are outdated.

Simulation. The link rate is selected as per standard IEEE 802.11a i.e 54 Mbps. Channel switching time is 200 us. In simulation, each node has saturated traffic directed to one of its neighbor. The simulated multi-hop configuration is used where 16 nodes are placed in a 4*4 grid where grid unit is 0.7 communication range. The communication range of each node is 250m. Figure below gives aggregate throughput vs channel hop in multi-hop environment.

Figure below also shows aggregate throughput vs packet length. The data is compared with CAM-MAC protocol.

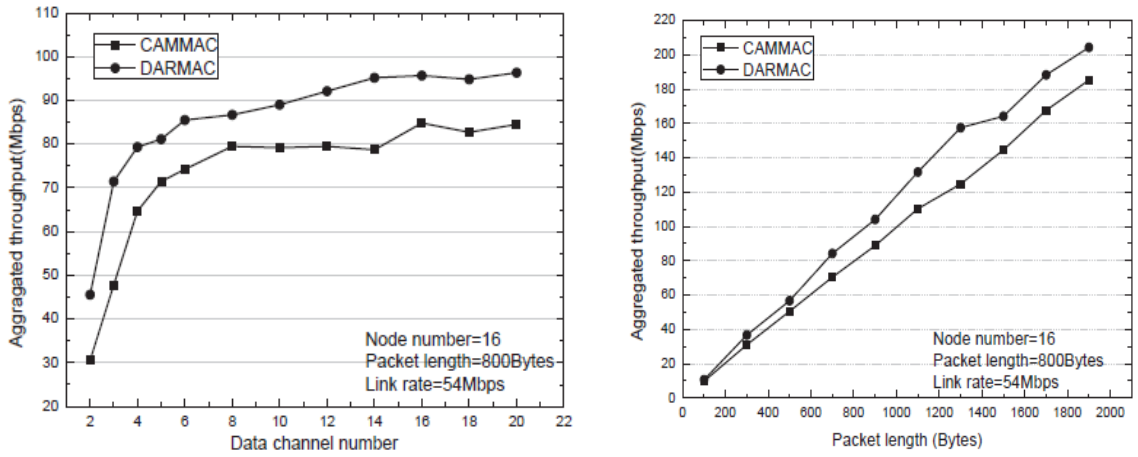


Fig 26. DARMAC vs CAM-MAC

The proposed DARMAC protocol improves CAM-MAC by introducing a new control channel handshake. The new control channel handshake not only is more efficient by eliminating a cooperation avoidance period in CAMMAC, but also allows nodes to reserve data channel before it is available.10-20% performance gain has been achieved by DARMAC. Further, the lower link rate gives better performance because the transmitter and receiver pairs will stay more time in the data channel, which equivalents to less control channel handshake. The increased packet length not only improves data channel efficiency itself, but also reduces control channel collisions by taking more time in data channel transmitting data packets. DARMAC has used high data link rate (54 Mbps), however it was implemented on a 4*4 gird but its performance is better than CAM-MAC.

3.3 H-MMAC

Protocol Description. H-MMAC stands for Hybrid Multichannel MAC protocol which is a combination of dedicated control channel and split phase (beacon intervals for synchronization) category protocols. This protocol is designed by Kyung Hee University Korea in 2012. The H-MMAC protocol allows nodes to transmit data packets while other nodes try to negotiate the data channel during ATIM (beacon) window which is not feasible in split phase protocols. The assumptions are as follows:-

- There are N non-overlapping channels which can be used. The beacon interval is divided into 2 sub-intervals: ATIM window, data window. One channel is defined as a default channel (CH1) just in ATIM window. The default channel is used to transfer data packets like other channels outside the ATIM window.
- Nodes have prior knowledge of how many channels are available.

Each node has a single half-duplex transceiver which is capable of switching the channel dynamically.

- All nodes are time-synchronized and operate the IEEE 802.11 DCF mechanism.

Each Node maintains a NIL (Neighbor Information List) and Preferred Channel List (PCL). The NIL stores the state, type and transmission mode (Tx mode) of neighbor nodes. The PCL stores the state of every channel and how many node pairs already reserved this channel. There are 2 states of a neighbor node: Idle and Busy. Idle node do not exchange data packets in the current beacon interval. Busy state indicates the node will exchange data packets in the current beacon. Normal transmission (N-Tx) is the transmission performed within data window. By using the next ATIM window for data transmission, Extra transmission (E-Tx) is longer than Normal transmission. Each node can choose either one of the transmission modes according to the number of packets in its buffer (Pkt Threshold) and the number of nodes allowed to transmit during ATIM window (ExtraTx Threshold).

The type of neighbor node can be one of 4 types: Normal, Ongoing, Limited and Unknown. Normal nodes are the nodes that do not lose any control messages from their neighbors. Nodes which are exchanging data during the ATIM window are classified as Ongoing nodes. Limited nodes are the nodes which lost information of some neighbors because they were busy with data transmission in the last ATIM window. If a node does not know any information of its neighbor node, the neighbor node is an Unknown node. The way to find the type of node A's neighbor node at the start of the third ATIM window is shown as an example in Fig. 27. Node B is a Limited node because it was Ongoing node in the last ATIM window. Node C is an Unknown node because node A lost node C's ATIM messages in the last ATIM window. But in the node D's point of view, node C is an Ongoing node and node G is a Normal node. If the neighbor node uses E-Tx mode from the beacon 1, its type is changed to Ongoing, Limited and Normal in the ATIM window of beacon 2, 3 and 4 respectively in the NIL.

Node A updates its type itself (Normal or Ongoing) and then updates its NIL before each beacon as the Table III. Whenever node A overhears ATIM messages from node j, the State changes from Idle to Busy and Tx type is updated to corresponding transmission mode of node j. The PCL is updated when the node overhears ATIMACK/ ATIM-RES messages or when the node selects a channel to use in data window.

• All the channels are reset to Idle state at the start of each beacon interval.
• If node A selects a channel to exchange data, this channel is changed to Selected state.
• When node A knows that its neighbor will use channel j through ATIM-ACK/ATIM-RES, it changes the state of that channel from Idle to Busy and increases the counter by one.

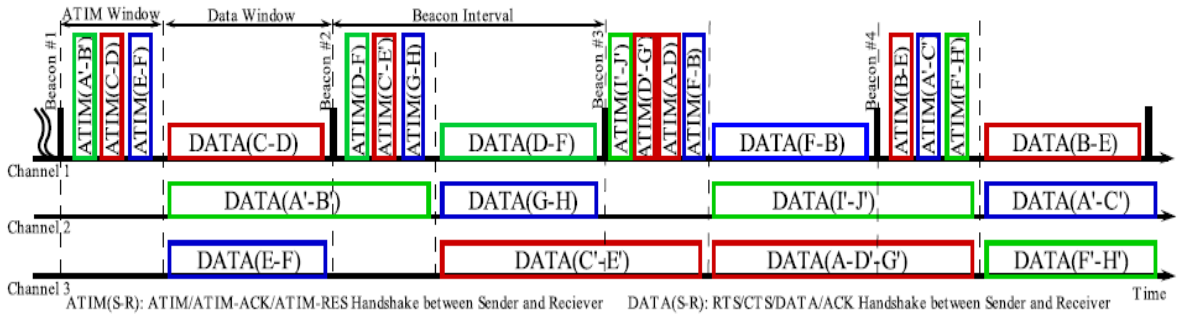


Fig 27. Operation of H-MMAC protocol

TABLE I
NODE A'S NIL

Node	State	Type	Tx mode
B	Idle	Limited	N-Tx
C	Idle	Unknown	N-Tx
D	Busy	Normal	E-Tx
E	Idle	Unknown	N-Tx
...	...	...	...

TABLE II
NODE A'S PCL

Channel	State	Counter
1	Idle	0
2	Busy	1
3	Selected	2

TABLE III
NIL'S UPDATE

Node A's Type	Before update			After update		
	State	Type	Tx mode	State	Type	Tx mode
Normal	-	Unknown	N-Tx	Idle	Normal	N-Tx
Normal	-	Unknown	E-Tx	-	Unknown	N-Tx
Normal	Idle	Limited	E-Tx	Idle	Normal	N-Tx
Normal	Busy	Ongoing	E-Tx	Idle	Limited	E-Tx
Any	Busy	Normal	E-Tx	Busy	Ongoing	E-Tx
Ongoing	-	Ongoing	-	-	Unknown	E-Tx
Normal	Busy	Normal	N-Tx	Idle	Normal	N-Tx
Any: Normal or Ongoing type			Ongoing: node that is not Ongoing type			

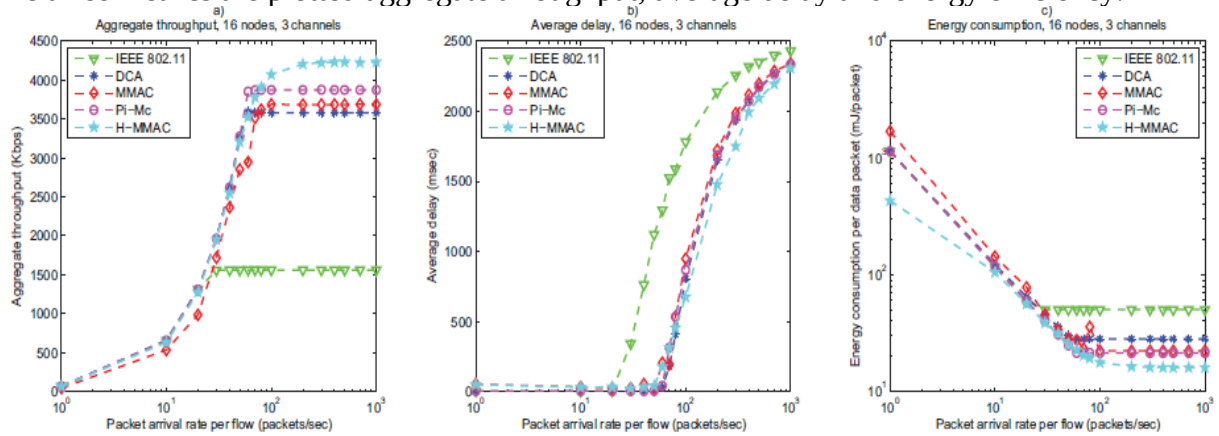
The operation of H-MMAC protocol is as follows:-

(a) If a node has data to send, it checks the receiver's type in its NIL. If the receiver's type is Ongoing or Unknown, it has to wait for next beacon to try again.
(b) Based on the Pkt Threshold and ExtraTx Threshold, the sender decides which transmission mode is used.
(c) The sender attaches its PCL and transmission mode into ATIM packet and sends to the receiver.
(d) Upon receiving ATIM, the receiver selects the best channel from its PCL. Then the receiver sends ATIM-ACK indicating the selected channel to the sender.
(e) The sender sends ATIM-RES to confirm the data channel selected by the receiver.
(f) After the ATIM window, the sender and receiver switch to agreed channel for exchanging data.

Simulation. In simulation, the paper has compared all categories of MAC protocols used for MANETs with H-MMAC. The scenario used is 16 nodes, 3 channels in an area of 250*250m. Each simulation was performed for 5 seconds and the simulation results are an average of 30 runs. The rest parameters are given in the table below:-

Transmission range	250 m
Data rate	2 Mbps
Data packet size	512 bytes
Beacon Interval	100 ms
ATIM window	20 ms
SIFS / DIFS /Slot time	16 μs / 34 μs / 9 μs
Retry limit	4
Channel switching time	224 μs
Transmit/Receive power consumption	1.65W / 1.4W
Idle/Doze power consumption	1.15W / 0.045W
Pkt_Threshold	20 packets
ExtraTx_Threshold	1 node

The three metrics are plotted aggregate throughput, average delay and energy efficiency.



Further, it was also found using analytical methods that H-MMAC also equivalent Dynamic TDMA which was used by DRDO as shown below.

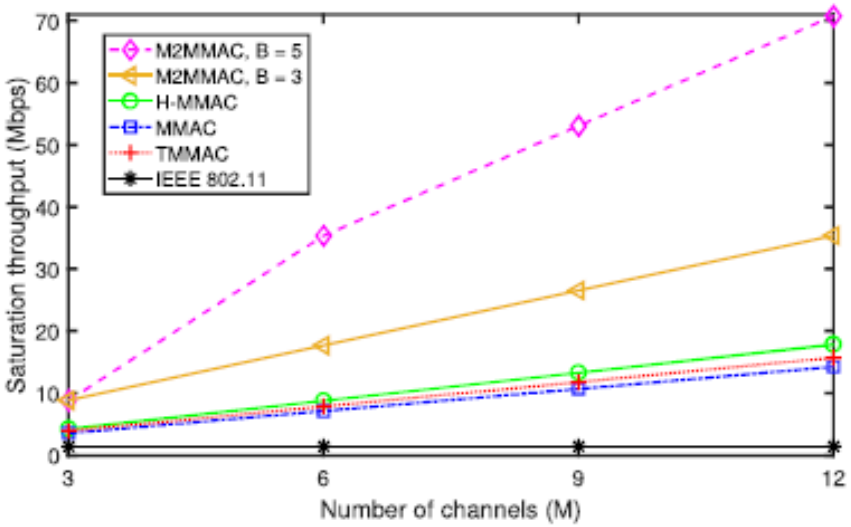


Fig 27. Simulation & Analytical results of H-MMAC

H-MMAC is a hybrid protocol which combines advantages of two main categories of MAC protocols used from MANETs i.e Dedicated Control Channel category and Split Phase (Beacon intervals for synchronization) category.

3.4 TMMAC

TMMAC is TDMA based multi-channel MAC protocol and is from the family of split-phase MAC protocols. It is a traffic adaptive and energy efficient scheduling algorithm. In addition to conventional frequency negotiation, TMMAC introduces lightweight explicit time negotiation. In TMMAC, time is divided into fixed periods, which consists of an ATIM (Ad Hoc Traffic Indication Messages) window followed by a communication window. The ATIM window size is dynamically adjusted based on different traffic patterns to achieve higher throughput and lower energy consumption. The communication window is time slotted, each of which is called a time slot. The duration of each time slot is the time needed for a single data packet transmission or reception. During the ATIM window, each node decides not only which channels to use, but also which time slots to use for data communication. Then each

node adopts the negotiated frequency for each time slot to transmit or receive data packets. Its main features are listed below:-

(a) TMMAC avoids contention based communication for data packets in the communication window and allows nodes to use different frequencies within different time slots through two dimensional negotiation. This property allows TMMAC to achieve more efficient bandwidth usage. Further, TMMAC achieves aggressive power savings by putting a node into doze mode in a time slot whenever it is not scheduled to transmit or receive a packet. Finally, TMMAC supports broadcast very efficiently.

(b) It dynamically adjusts the ATIM window size efficiently based on different traffic patterns which improves both the network throughput and energy efficiency of TMMAC.

Design. In TMMAC, time is divided into fixed-length beacon intervals and each beacon interval is comprised of an ATIM window and a communication window. Different from 802.11 PSM and MMAC, in TMMAC, the ATIM window size is dynamically adjusted and the communication window is further divided into time slots. During the ATIM window, all the nodes listen to the same default channel for negotiation. Four types of messages are used for negotiation: ATIM, ATIM-ACK (ATIM-Acknowledgement), ATIM-RES (ATIM-Reservation) and ATIM-BRD (ATIM-Broadcast). They are called ATIM control packets. In TMMAC, the communication during the ATIM window is contention based and uses the same scheme as the one used in 802.11 DCF. During the negotiation, the sender and receiver decide not only which channels to use, but also which time slots to use for a set of data packets, the number of which is specified by the sender. Then in each time slot, each node adopts the negotiated frequency to transmit or receive data packets. The duration of each time slot is long enough to accommodate a data packet transmission, including the time needed to switch the channel, transmit the data packet and the acknowledgement.

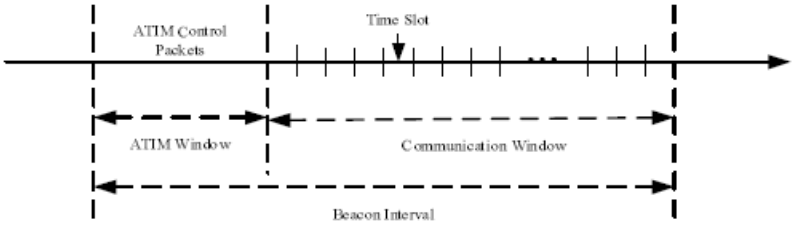


Figure 28: Architecture of TMMAC

Two data structures are used in TMMAC. The CUB (channel usage bitmap) is the main data structure that needs to be maintained at each node. Each CUB represents the current usage information of one channel. So if the radio transceiver has M available channels, there are M CUBs in each node. These CUBs are used to keep track of the allocations of all the previous negotiations in the current ATIM window. The second data structure is CAB (channel allocation bitmap). CAB describes which time slots in that channel are allocated by the current negotiation.

The maximum throughput in TMMAC is achieved only when the optimal ATIM window size is used. If l_{atim} is different from l_{opt}, it results in bandwidth waste either in the ATIM window or in the communication window. However, there is no fixed l_{opt} in TMMAC which is able to achieve the maximum throughput under all situations. In TMMAC, each node adjusts its

ATIM window size dynamically, allowing different nodes to have different ATIM window sizes. We use a finite set of ATIM window sizes {ATIM1; ….ATIMi; ATIMi+1; …ATIMm}, in which ATIM1 is the minimal ATIM window size, ATIMm is the maximal ATIM window size, and ATIMi+1- ATIMi = lslot. To avoid collisions between ATIM control packets and data packets in the default channel, the default channel is never used for data communication in the time slots before ATIMm. However, other channels can be used for data communication in these time slots as long as they are not within a node's current ATIM window. When a node is sending an ATIM control packet, it piggybacks its ATIM window size for the next beacon interval. Thus, the neighboring nodes know its ATIM window size. There are two possibilities when node A wants to send a packet to node B. If node A knows node B's ATIM window size, node A decides whether the negotiation can be finished within min{A's ATIMwindow; B's ATIMwindow}. If yes, node A sends the ATIM packet to node B. Else, node A waits for the next beacon interval. If node A does not know node B's ATIM window size, node A decides whether the negotiation can be finished within ATIM1. If yes, node A sends the ATIM packet. Else, node A waits for the next beacon interval.

After deciding whether the network is saturated, the corresponding rules are applied. If the network is saturated, it means all the available bandwidth in the communication window is scheduled for data communication. If yes, i.e., Pschedule >= Paccommodate, we decrease the ATIM window size by one level to leave more bandwidth for data communication. If not, i.e., Pschedule < Paccommodate, we increase the ATIM window size by one level to leave more bandwidth for negotiation. If the network is not saturated, we decrease the ATIM window size by one level to save more power. There is a special case in which a node does not adopt the ATIM window size computed based on the above rules. If a node does not get the opportunity to broadcast its current ATIM window size in the last beacon interval, i.e., no node knows its current ATIM window size, and it does not have any packets to send in this beacon interval, this node resets its current ATIM window size to ATIM1.

Simulation. For simulation NS-2 simulator is used. 80 nodes are randomly arranged in an area of 1 sq km. The Communication range for each node is 250 mtrs and 500 mtrs is the CSR. The ATIM window can vary from 8.57 ms to 31.43 ms. Three channels each with 2 mbps data rate are used and first channel is common channel used for establishing data communication on other channels. The protocol performance is compared with 802.11 DCF and MMAC protocol. TMMAC achieves 113% more aggregate throughput then MMAC and 4.5 times than that of 802.11 MAC.

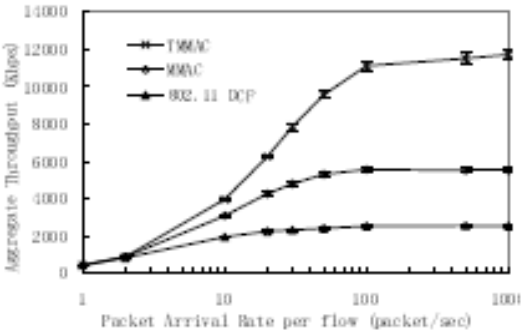


Figure 29: Aggregate Throughput Comparison

CHAPTER 4

EFFICIENT MAC PROTOCOLS FOR LARGE SCALE MANETS

Based on literature survey & outcomes from chapter 2 & 3, the following MAC protocols are found to be suitable for supporting a large scale MANET.

4.1 C-DTSAP

C-DTSAP is cluster based dynamic TDMA slot assignment protocol for large scale MANETs. It was designed by Beijing University in 2019. C-DTSAP utilizes proper network structure and clustering technology, and realizes the management of large scale networks by clustering. In addition, the proposed protocol makes high-effective use of slot resources by the slot assignment process to reduce delay and improve time slot reuse ratio.

USAP (unifying slot assignment protocol), the classic schedule-based MAC protocol, is the basis of C-DTSAP. It allows nodes to choose slots from the unscheduled slots among its neighbor nodes, coordinates the announcement and confirmation of slot assignment within the two-hop range, and ensures no conflict after assignment. C-DTSAP utilizes clustering structure to divide large network into clusters, which makes C-DTSAP run independently among clusters without inter-cluster interference. As depicted in Fig. 31, each cluster works in a unique frequency. In addition to normal nodes, cluster-head nodes and gateway nodes also play an important role in the network.

(a) The **cluster-head node** in each cluster is pre-assigned and is responsible for intra-cluster management. The distance between other nodes in the cluster and the cluster-head node can be multi-hop. It can determine the gateway node for its own cluster as well.

(b) The **gateway node** plays a key role in cluster-to-cluster communication. In each cluster, a gateway node is selected to communicate with another cluster. That means each of the two clusters has its own gateway node and there are two gateway nodes for communication between two clusters. The gateway nodes switch their frequency from one to the other and only the gateway nodes can work in two clusters.

(c) All nodes in the cluster are fair. Each node can dynamically apply for slot resources according to its traffic loads.

.

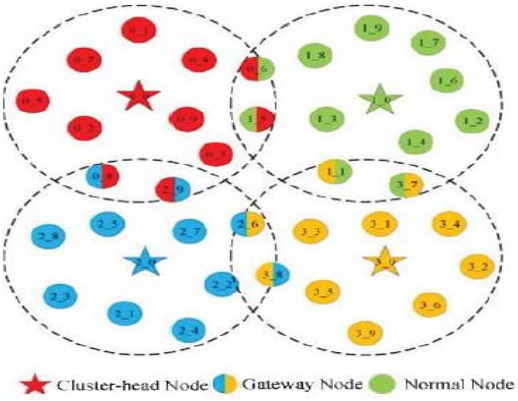


Figure 30: Cluster Network Topology

In C-DTSAP, there are three kinds of control packets, NMOP0, NMOP1 and NMOP2. With these control packets, nodes can get enough information to schedule slot assignment, for ensuring transmit data packets without conflict.

(a) *NMOP0:* The function of the NMOP0 is neighbor discovering. Each node can establish a neighbor table that includes the information of one-hop neighbors and two-hop neighbors with the NMOP0 interactions. If the NMOP0 sent by a neighbor is not received within a certain time, the information about this node in the neighbor table will be deleted.

(b) *NMOP1:* The NMOP1 mainly contains the data slots that are required to announce to neighbor nodes and the status of the data slots. Each node calculates the required number of data slots based on their traffic loads, and encapsulated into the NMOP1.

(c) *NMOP2:* The NMOP2 is transmitted for the confirmation of the NMOP1. It mainly contains reply information to neighbors. To avoid collisions, the nodes within a two-hop range could not occupy the same data slots.

C-DTSAP utilizes channel segmentation to divide the channel into *F* frequencies. Here, *F* represents the number of frequencies (i.e. the number of clusters). As depicted in Fig. 32, each frequency is used by each cluster. There are several frames composed of four sub-frames in one frequency. The four sub-frames are:

(a) *Detection sub-frame:* This sub-frame is to select the gateway node of each cluster. The detection sub-frame consists of *M* detection slots. Here, *M* represents the number of clusters. At the beginning of that, all nodes from different clusters switch own current working frequency to the public frequency. Then each cluster takes turn to occupy a detection slot to broadcast detection packets. If a node can receive detection packets from other clusters, it will be marked as a candidate gateway. Before the end of this subframe, all nodes switch the public frequency to their own working frequency.

(b) *Control sub-frame:* This sub-frame is mainly for the interaction of NMOP0s and the NMOP1s. The control subframe consists of *N+M* control slots. Here, N represents cluster receiving threshold (i.e. the number of nodes in each cluster). M represents the number of clusters. Each node in its own cluster is assigned a control slot by the cluster-head. The rest control slots are reserved for gateway nodes from own and other clusters. Intra-cluster nodes take turns to broadcast NMOP0s and NMOP1s in own occupied control slots. The gateway node only work in a cluster during this control sub-frame. When next control sub-frame arrives, gateway nodes could work in the other cluster.

(c) *ACK sub-frame:* This sub-frame is to confirm the slot assignment. Similar to the control sub-frame, the ACK subframe also has *N+M* ACK slots. Each node in cluster is assigned an ACK slot. In this sub-frame, nodes transmit NMOP2s to each neighbor node.

(d) *Data sub-frame:* This sub-frame is to transmit data packets. The number of data slots is determined by scale of the network. Each node can use two data slots even If there is no data to transmit.

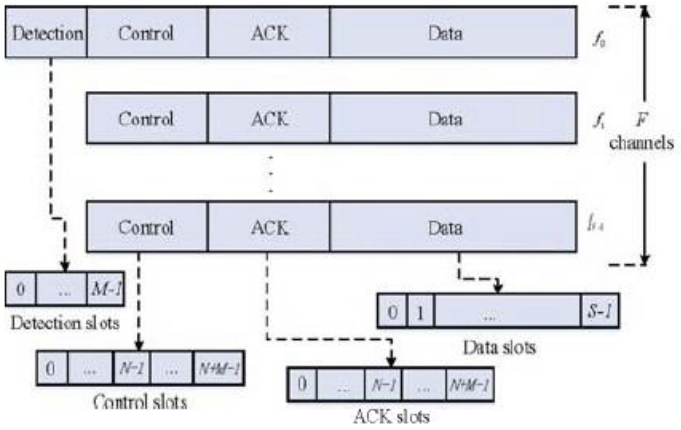


Figure 31: Frame format of C-DTSAP

Slot Assignment. In C-DTSAP, all nodes assign data slots following a rule that slot reuse outside the two-hop range. As depicted in Fig. 33, each node estimates the traffic loads and calculates the number of slots for a neighbor. After all neighbors are accumulated, the quantity of slots can be calculated. Then each node determines to need another idle slots or release some assigned slots, according to the number of slots assigned in the last slot assignment.

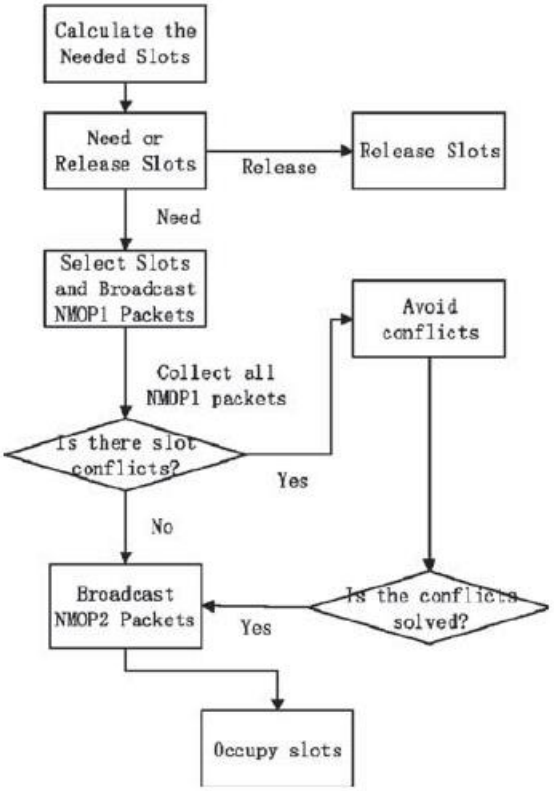
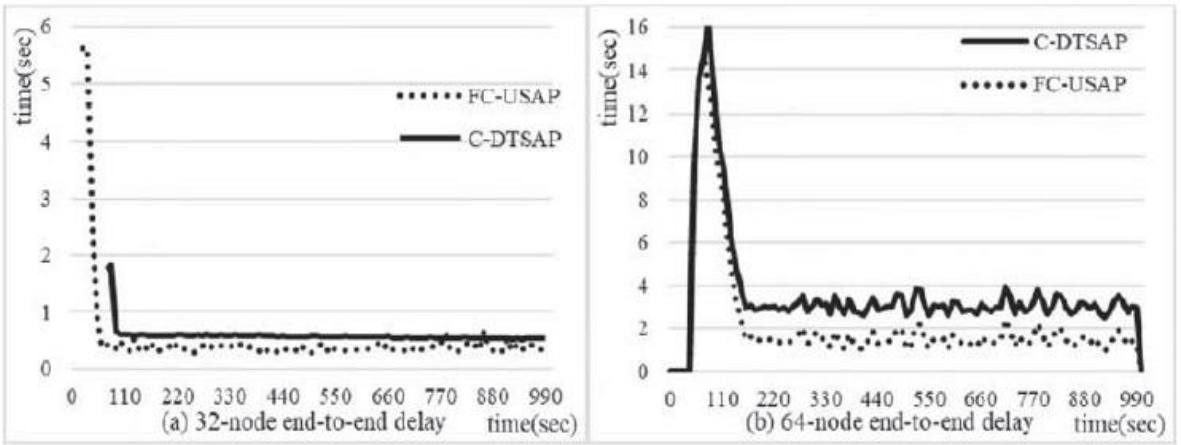


Figure 32: Slot Assignment Process

If the node *i* needs another new slots, it selects the required slots from the unassigned slots, then broadcasts announcement through an NMOP1 to neighbors in its control slots, and receives the NMOP1s from its neighbors in the rest control slots. After the control sub-frame ends, the node *i* generates or updates its own local data slot table. If the node *i* is a gateway node, it has two data slot table (i.e. the local slot table and slot table of the other cluster),

because it works alternately in two clusters. The data slots has four states: 00, 01, 10 and 11. Here, 00 is idle state, 01 is self-occupied, 10 is occupied by one-hop neighbor, and 11 is occupied by two-hop neighbor. All nodes only can announce the slots with status 00 to the neighbors. After collecting the NMOP1s from all neighbors, the node *i* checks if the announced slots will cause a conflict. When the announcement conflict occurs from neighbors, the node *I* replies the confirmation to the node with smallest ID to allow it to occupy the slot through NMOP2 in its own control slots. The node *i* receives NMOP2s from the neighbors in the other control slots and inserts that into the local NMOP2 buffer queue. Then the node *i* acquire the number of the neighbors from its own neighbor table. In most instances, only if the number of NMOP2s is equal to the number of neighbors, and this node receives all the confirmation from the neighbors, this slot can be assigned. Then the status of the assigned slots are updated and broadcasted to neighbors in its own next control slot.

Simulation Results. The experiment uses OPNET Modeler 16.0 for modeling and simulation, generates different scales' network topologies, configures parameters of each node and compares the performance of C-DTSAP with FC-USAP. In the simulation experiment, there have 32-node, 64-node, 96- node and 128-node scales and each network has 4 clusters. Simulation time is set to 1000s and the number of data slots is 152. The range of each scenario is within 15*15 km square and the communication range of each node from 1.0 km to 1.5 km. The network layer uses the Optimization Link State Routing Protocol (OLSR) and the physical layer uses rate adaptation. Each scenario loads two traffic flows and each traffic is set to 100 Kbits/s. In this section, we test C-DTSAP in different scenarios and compare end-to-end delay, traffic received rate and slot reuse rate with FC-USAP. As shown in Fig.34, the end-to-end delay comparison of the two protocols from the 32-node network to the 128-node network. The end-to-end delay is obtained according to the simulation time of sending and receiving packets. In the beginning of the simulation, the network is in the initial stage and all nodes have not started receiving packets. Therefore, the delay is very high in this process. In the 32-node network, when the network converges, the delay decreases and remains stable around 0.5s. There is no significant difference in the delay of the two protocols in small-scale networks. In the 64-node network, the two delay curves are stable, but the delay of FC-USAP is slightly higher than that of C-DTSAP. With the scale of the network increases, the convergence time of FC-USAP becomes longer. As depicted in Fig.34 (d), in the 128-node network, the delay jitter of FC-USAP is large and unstable, while the delay of C-DTSAP is stable and remains at 2.5s even in the case of end-to-end more than 15 hops. Under the same number of transmission hops, the delay of FC-USAP is much higher than that of C-DTSAP.



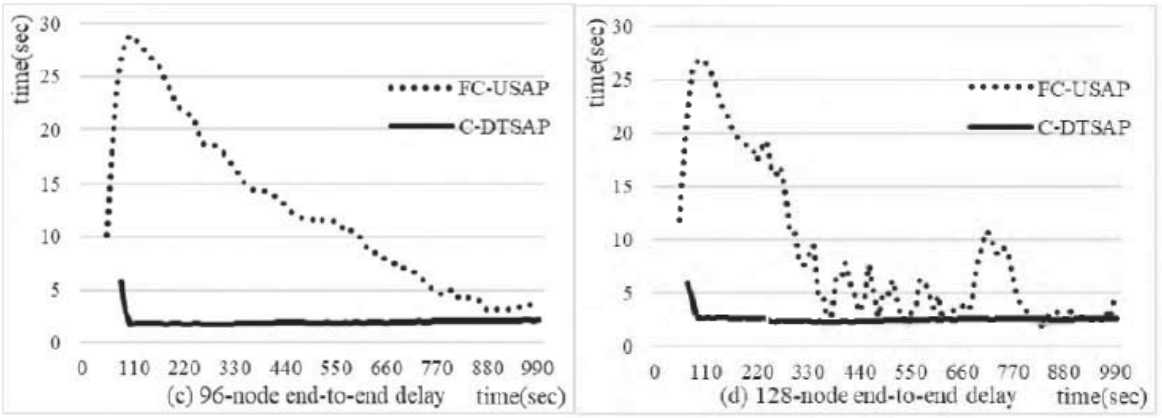


Figure 33: End to end delay

The technique explained above is found to be suitable for large scale MANETs with minimal end to end delay.

4.2 Hybrid MAC protocol

This approach is a hybrid technique which makes use of both CSMA and TDMA based MAC protocol. This protocol exploits CSMA based MAC protocol explained for small scale MANETs for intra-cluster communication and Dynamic TDMA for inter-cluster communication. The frame structure for the same is shown below:-

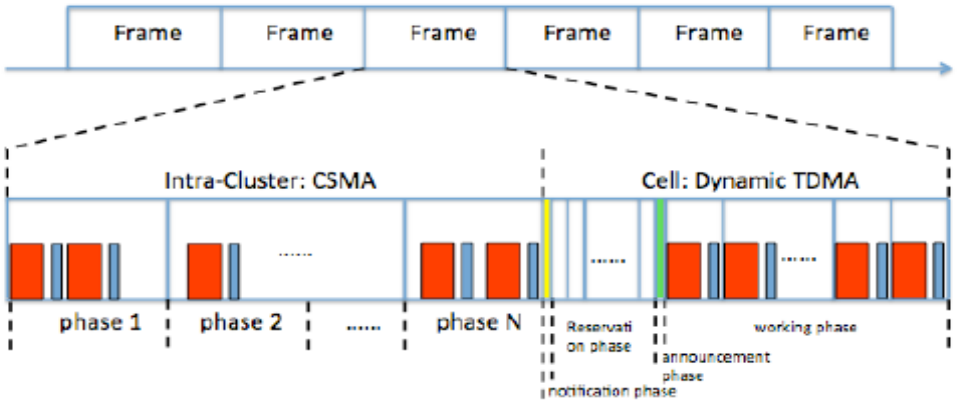


Figure 34: Hybrid MAC Protocol

The former is further split into *N* phases, in each of which a part of cluster members send their packets in CSMA manner. The packets are gathered at cluster heads, which are going to be sent in the dynamic TDMA period. This approach is less power efficient than dynamic TDMA.

4.3 Clustering Techniques

The various clustering approaches are tabulated below for large scale MANETs.

APPROACH :1	**APPROACH :2**	**APPROACH :3**
➢ Intra-cluster: CM to CM: P2P CM to CM via CH	➢ Intra-cluster: CM to CM : P2P CM to CM via CH	➢ Intra-cluster: CH -Centralized Control (routing & control decisions by cluster head only)
➢ Inter-cluster: CM to GN directly CM to GN via CH	➢ Inter-cluster: CM to GN via CH	➢ Inter-cluster: CH to CH via GN
▪ Slot Decisions at node level ▪ Slot Allocation based on acknowledgment from two hop neighbors ▪ Control slots allocation by CH	▪ Control messages exchange between one hop neighbors only ▪ Less Control Overhead then approach 1	▪ Control , scheduling & routing decisions by CH ▪ Requires continuous scheduling calculation ▪ Increased Control overhead & less scalability ▪ Single point failure if control node fails ▪ Better performance then approach 1 & 2 , but not optimal for changing topology & traffic conditions

CHAPTER 5

MARKET SURVEY

An Indian market survey was undertaken to study the technical specifications of SDRs available in market. The details are tabulated below:-

M/s Exicom & M/s BrijSystems (Mbi)	**M/s Sankhya Labs (Bengaluru)**
(i) Mesh topology (ii) 1.5 Kms Single hop (iii) Freq- 1.14 to 1.50 GHZ (iv) B/w – 1.25 MHz (v) Max rate – 12 Mbps (vi) Total output power-2W (vii) Nodes – 35 (viii) Unique Token based Channel Access Mechanism (Final Trials Stage)	(i) VHF/UHF Range Narrow Band- 16 nodes Wide Band – 32 nodes (ii) Data rate – 28 kbps (narrow band) 2 Mbps (wide band) (iii) Single & multicarrier ofdm (iv) B/w- 1Khz to 8 Mhz (v) Mesh Topology (vi) Range – 1.5 sq kms

CHAPTER 6

LARGE SCALE MANET SIMULATION IN NS3

A large scale MANET has been simulated on NS-3.27 with 100 nodes in an area of 25 Sq Kms. The area is divided into 4 clusters. A single channel of 54 Mbps is segmented into 4 narrow band sub-channels for 4 clusters. A hybrid MAC is installed on all nodes for communication namely CSMA based E-MAC protocol for Intra-cluster communication and TDMA based MAC for inter-cluster communication. The clustering approach 1 defined in section 4.3 has been implemented. The simulation parameters are listed below:-

(a)	Nodes	-	100
(b)	Area	-	25 sq kms
(c)	MAC protocol	-	Hybrid MAC
(d)	Routing Protocol	-	OLSR
(e)	Channel rate	-	54 Mbps
(f)	Packet size	-	1000 bytes
(g)	Intra-cluster	-	02 hops
(h)	Inter-cluster next cluster	-	04 hops
(i)	Inter-cluster diagonal cluster	-	05 hops
(j)	SIFS	-	6 us
(k)	Traffic rate	-	100 kb/sec

NS-3 has a tool to create and animate the scenario designed known as Net-Anim 3.108 on which the initial & hoping scenario has been simulated. The screenshot of the same is shown in next page. Post simulation, the packet capture files are created which are captured by a NS-3 tool Wireshark. The snapshot of the same has been placed in subsequent pages below. The delay captured from these packet capture files was then captured in simulation in tool GnuPlot for extracting simulation time vs end-to-end delay plot. The delay achieved is summarized in table given below:-

Scenario	**Delay(ms)**
Intra-cluster - 2 hops	21
Inter-cluster (Next cluster) – 4 hops	70
Inter-cluster(Diagonal cluster) – 5 hops	81

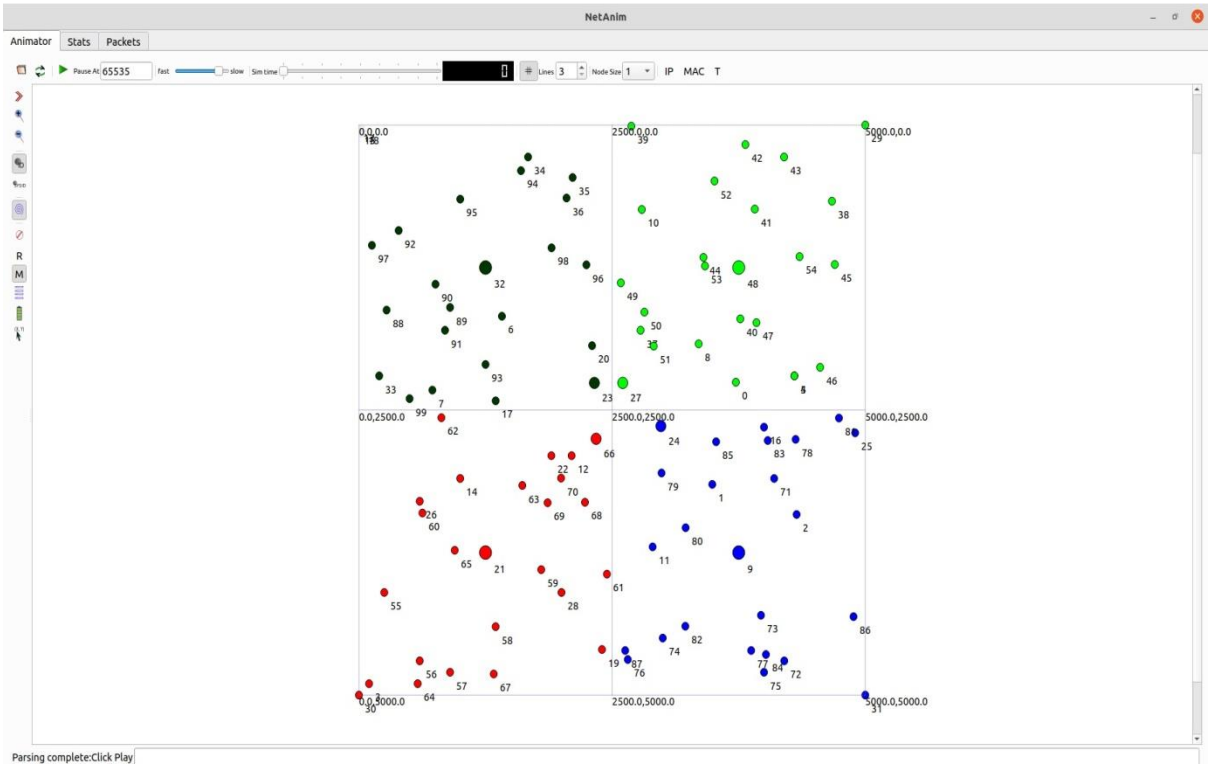


Figure 35: Initial Simulation Scenario with 100 nodes in an area of 25 sq kms

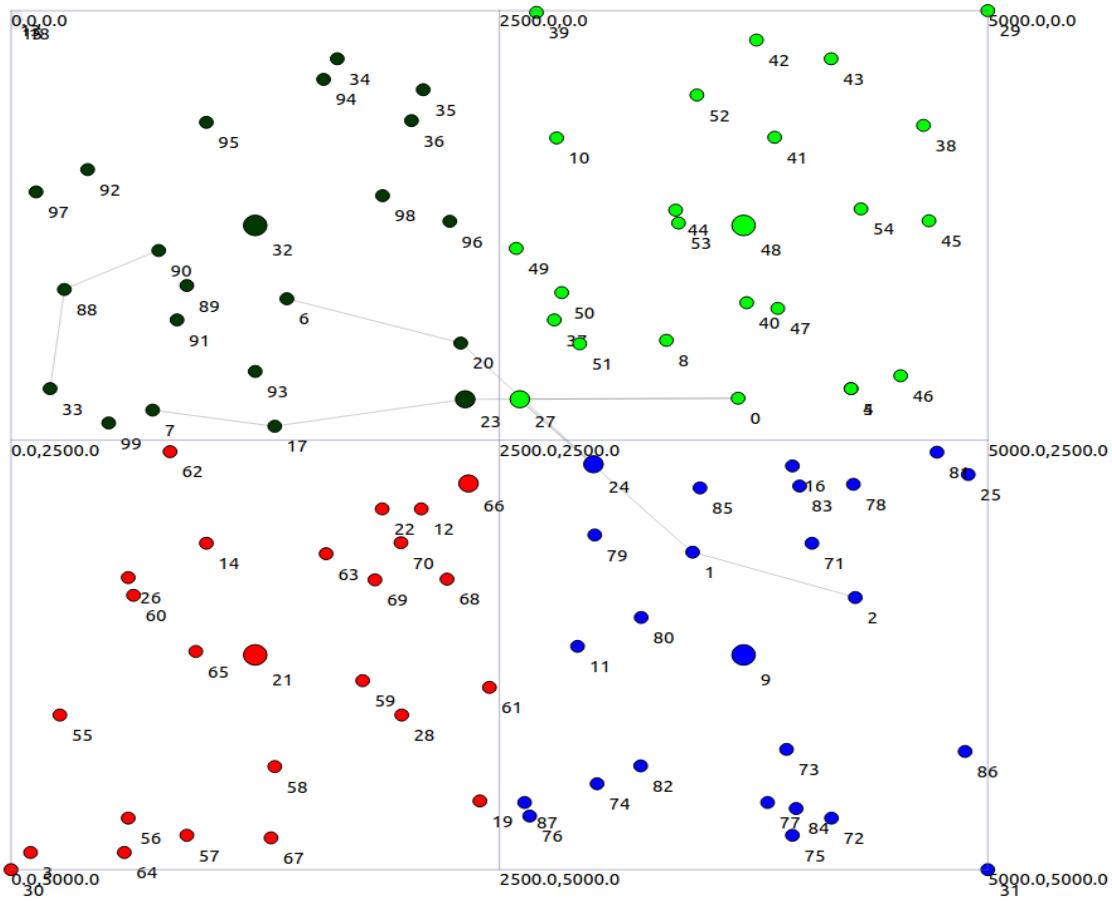


Figure 36: Hoping Scenario depicting intra-cluster & inter-cluster communication

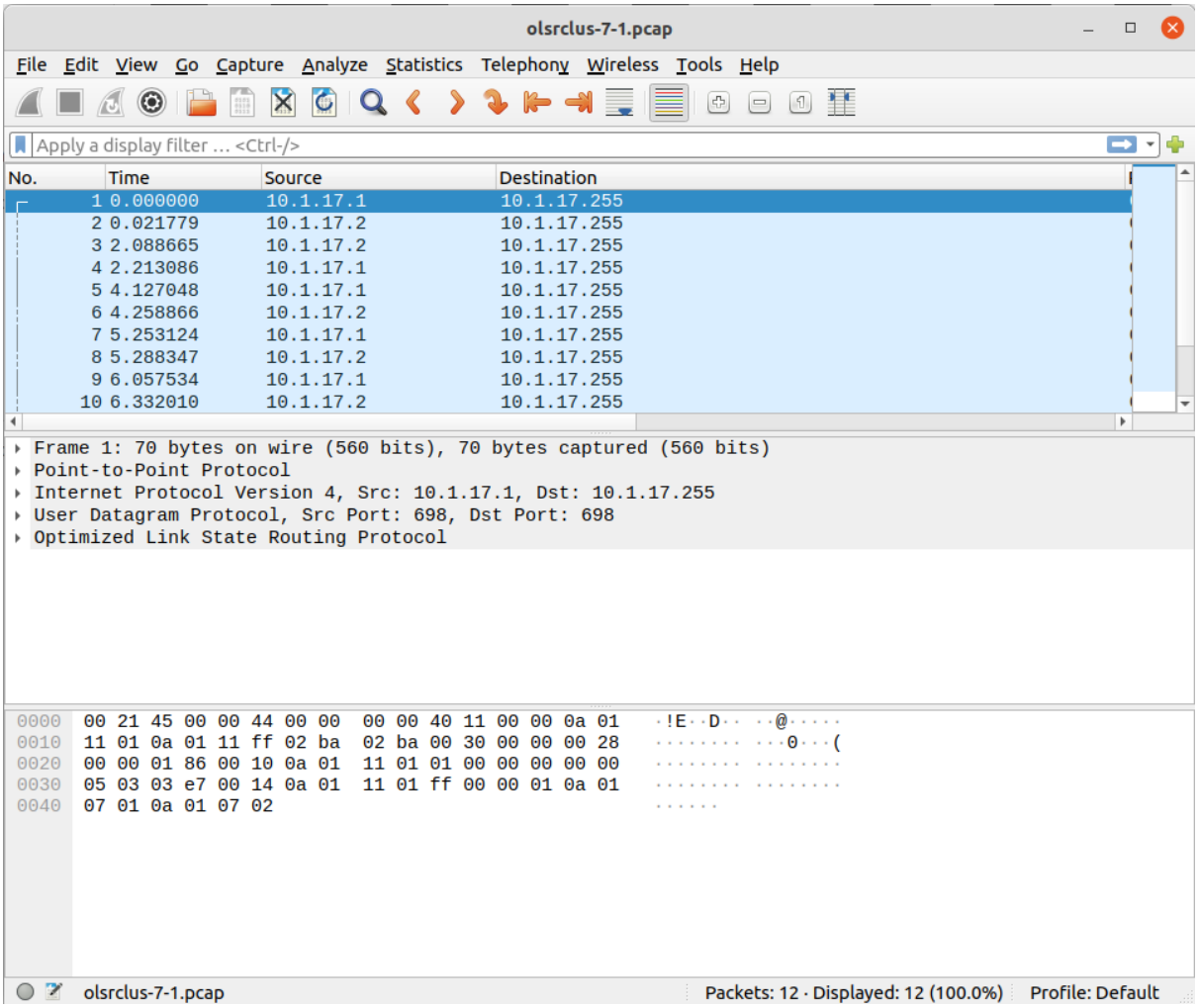


Figure 37: NS3 Wireshark packet capture file

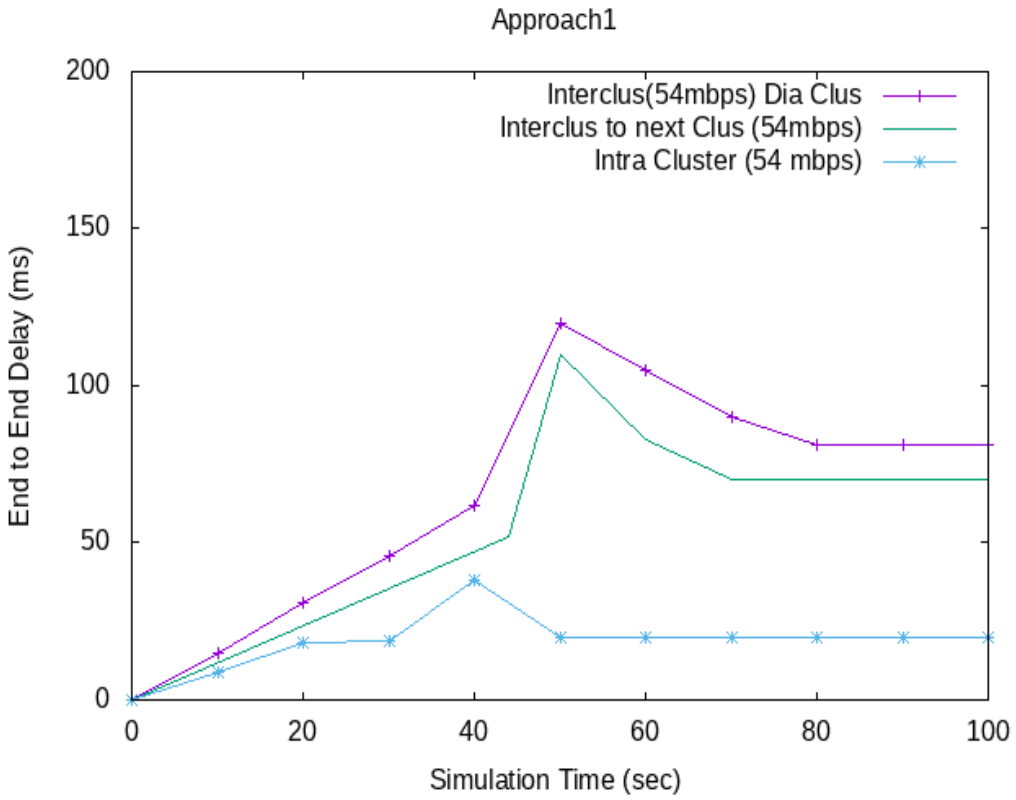


Figure 38: NS3 Gnuplot of Simulation time vs End-to-End Delay

CHAPTER 7

CONCLUSION & FUTURE SCOPE

Based on the above survey, it has been found that for small scale MANETs, Dedicated Control Channel MAC protocols such as CAM-MAC, DARMAC can be used. Further, in split phase category Dynamic TDMA – TMMAC and in hybrid category H-MMAC provides optimum performance. Exploiting multi-channel approach in CSMA and TDMA based approach provides more bandwidth and increased throughput. Therefore, for small scale MANETs Multi-channel MAC protocols provide optimum performance.

For Large scale, MANETs clustering techniques are to be exploited in order to convert large area in small size clusters according to approaches specified in section 4.3 as per characteristics of the MANET. The next step is to choose appropriate MAC protocol for intra-cluster and inter-cluster communication. Dynamic TDMA and Hybrid MAC algorithms are most suitable for large scale MANETs. The end-to-end delay achieved in simulation is optimum for distr, however the centralized cluster head approach provides better performance then distributed approach.

The Future scope of the research is listed below:-

(a) Scalability study of scenario.

(b) Scenario study involving multiple nodes transmission and study of collision avoidance strategy.

(c) Introduction of mobility in scenario.

(d) Introduction of FH in MANET.