Power Control in MAC Layer of Wireless ad-hoc Networks

Hemant Kumar Rath(04407001)
Guide: Prof. Abhay Karandikar

Abstract— **In this report, we mainly discuss the various methods of Power Control in Medium Access Control (MAC) layer of wireless ad-hoc networks. We have considered both Code Division Multiple Access (CDMA) based and non-CDMA based wireless ad-hoc networks in this report. We assume decentralised wireless ad-hoc networks throughout this report. We considered data rate, capacity optimisation, support for real-time traffic, multi-class and multi-hop communication and load vs. information rate balancing etc., as the yard sticks of comparison between the various techniques.**

Keywords— **MAC, ad-hoc networks, RTS, CTS, Hidden and Exposed Terminal**

I. Introduction

WIreless ad-hoc networks are inherently limited by battery power and bandwidth constraints. The MAC layer design for wireless ad-hoc networks plays an important role for effective transmission between nodes, required for effective utilisation of spectral resources and increasing life-time of battery power.

IEEE Std 802.11 [1] for wireless LANs is being used as the defacto MAC protocol in the current implementations. The MAC protocol in this is based on Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) mechanism. It uses Request-to-Send (RTS) and Clear-to-Send (CTS) control packets before data transmission to create a silence zone [2], i.e., to acquire a floor for its data transmission to start with. This is explained in Figure 1. Here node A wants to transmit data to node B. It uses fixed power mechanism to transmit data, i.e., it transmits at a fixed power level (maximum allowed power) irrespective of the distance between the transmitting and receiving nodes. The use of RTS and CTS also solve the hidden and exposed nodes problems of wireless ad-hoc networks. Figure 2 explains this problem, where node E is the exposed node and H is the hidden node for the data transmission between nodes A and B.

As the node density increases, the fixed power scheme does not perform well. In this case, the transmitting power level is not a function of distance between the transmitter and receiver and hence there is a wastage of radio resources. This also prevents the neighboring nodes to transmit during one's transmission. But, if we transmit at a minimum required power level, then we may acquire a smaller portion of the floor and hence there can be increase in throughput. Also, the battery power is less used. So, transmitting at a different power level is very-much required in an wireless ad-hoc networks to get better throughput, and also to increase the life-time of the battery power of the mobile node.

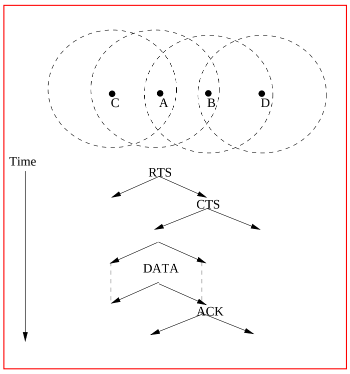

Fig. 1. RTS and CTS mechanism in IEEE Std 802.11

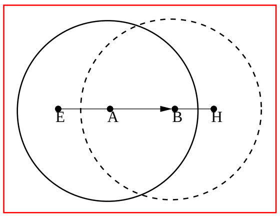

Fig. 2. Hidden and Exposed Terminals

In this report, we have investigated various methods of power control in MAC layer for both CDMA based and non-CDMA based wireless ad-hoc networks. Under the non-CDMA based category, we have discussed *Power Controlled Multiple Access (PCMA)* [3], *Power Controlled MAC (POWMAC)* [4] and *Dynamic Channel Assignment with Power Control (DCA-PC)* [5] techniques. Similarly, under the CDMA based category, we have considered *Interference Based Multiple Access (IBMA)* [6] and *Controlled Access CDMA (CA-CDMA)* [7] techniques in this report. All the above techniques are based on the following two basic requirements. They are: (i) guaranteeing a target transmission quality to each active link, at the actual interference level, and (ii) transmitting at a power level such that the other links' transmission quality is not suffered.

We have investigated their approaches for power control, the resource management techniques, data rates and throughputs, advantages and drawbacks as well as the open issues. All our comparisons are with respect to the IEEE Std. 802.11 [1].

II. Issues in Power Controlled MAC Protocols

The need of power control in MAC layer is to minimise the floor acquisition and hence to increase the effective utilisation of the radio resources. In any multiple access technique, any effective communication between a pair of nodes can be made, if the link between them are not affected by other, i.e., if, there are enough separation between the links are maintained, and a minimum Signal-to-Interference-plus-Noise Ratio (SINR) [8] as maintained at the nodes. This is valid for any multiple access technology.

At a radio receiver i, the SINR γ^i is given by,

$$\gamma^i = \frac{G^{ij} P_t^j}{\sum_{j=1, j\neq i}^{N} G^{ij} P_t^j + \eta^i} \tag{1}$$

Where, G^{ij} is the path path gain between nodes i and j, P_t^j is the power transmitted by the node j, η^i is the thermal noise power level at node i.

In CDMA based systems, the SINR includes a term relevant to the other link interference. In this perspective the collision avoidance mechanism is turned into an interference control one by means of power control technique.

For any power controlled MAC protocols, there must be a way to advertise the required SINR at a node for correctly receiving data. This is achievable, by using the following two principles [3].

Power Conservation Principle: Any transmitting node must transmit at a minimum power level, which is just sufficient for receiving at the specified receiving node.
Cooperation Principle: The node that commences a new transmission must not transmit loud enough to disrupt the ongoing transmissions between another pair of nodes.

A. Assumptions

In this report we assume the following three assumptions related to channel propagation model:

1. *Channel reciprocity holds*, i.e., the path gain between a pair of nodes is independent of direction of transmission.
2. *Channel stationarity holds*, i.e, the channel characteristics does not change during the course of control as well as data packet transmission.
3. *Channel uniformity holds*, i.e., the path gain for both control and data packets are same.

III. Non-CDMA based Power Control mechanism

A. PCMA

PCMA [3] a "variable bounded power" collision suppression mechanism and allows variable transmit power levels to maintain a SINR (γ_{th}) during data transmission on a per packet basis. PCMA maintains a desired SINR level, (γ_{des}) at the link establishment phase (floor acquisition phase) and γ_{th} during the data transfer phase.

A.1 Power Constraints in PCMA

PCMA defines the maximum and minimum transmission power of a node on the data channel and are specified by P_{t-Max} and P_{t-Min}. Also, it defines two thresholds for a receiving node. They are P_{RX-th} and P_{CS-th} for receiving the minimum received power level for receiving a packet and sensing the channel respectively. Now, for a successful data transmission between two nodes i and j the following constraints must be satisfied.
1. If, P_t^i is the power transmitted by node i to reach node j, then $P_{t-Min} \leq P_t^i \leq P_{t-Max}$
2. If, the received power at node j is P_r^j, then $P_r^j = (G^{ij} P_t^i) \geq P_{RX-th}$, where G^{ij} is the path gain between nodes i and j as defined before.
3. The observed SINR at any node j (γ^j defined by Eqn. 1) should be greater than the minimum required SINR, i.e., $\gamma^j \geq \gamma_{th}$.
4. It defines another term called "Noise Tolerance" E^k, for any receiving node k that is receiving any ongoing transmission in the neighborhood of a transmitting node i.

$$E^k = \frac{P_r^k}{\gamma_{th}} - \sum_{k=1, k\neq i}^{N} G^{ki} P_t^i + \eta^k \tag{2}$$

PCMA also defines another variable called $P_{t-bound}$. For a successful transmission $P_t^i \leq min_k(\frac{E^k}{G^{ik}}) = P_{t-bound}^i$ (based on cooperation principle).

A.2 PCMA Protocol Principle

Similar to IEEE Std 802.11, PCMA also relies on both (a) *collision avoidance* and (b) *collision resolution* mechanisms for successful communication between the transmitting and the receiving nodes [3]. Collision avoidance is based on the power control mechanism of PCMA, where as collision resolution is based on back-off algorithm. Similar to IEEE Std. 802.11, PCMA uses Request-Power-To-Send (RPTS) and Acceptable-Power-To-Send (APTS) handshake between a pair of transmitter and receiver to determine the minimal transmission power required for successful packet reception. The difference lies in that PCMA introduces a second channel, the busy tone channel, to advertise the noise tolerance level. During a data transmission period, each active receiver periodically sends a busy tone to advertise the maximum additional noise power it can tolerate. Any potential transmitter must first sense the channel for busy tones to determine the upper bound of its transmit power for a minimum time period (determined by the frequency with which the busy tones are transmitted). Actually, PCMA uses the signal strength of a received busy tone message to bound the transmission power of neighboring nodes. This way, power control mechanism is realised and spatial reuse is achieved. The assumptions in Section II-A are also valid in PCMA.

A.3 PCMA Protocol Steps

We have considered Figure 3 for the following discussions. Node i and j are communicating nodes and node i wants to start data communication with node j. Node k, l, s, u are neighboring nodes and potentially interfering nodes. Initially all nodes are in *IDLE* state and are monitoring the busy tone channels to determine their power

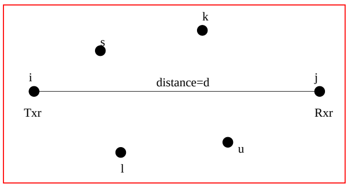

Fig. 3. Node placement

bound $P_{t-bound}$.
The protocol steps are explained below.

1. Node i wants to transmit data to node j. It monitors the busy tone channel and waits until $\alpha P^i_{t-bound} > P_{t-Min}$ and then goes for back-off. Where, α is a constant and its value is 0.9 (based on simulation results). At the end of the back-off, if, $\alpha P^i_{t-bound} > P_{t-Min}$ still holds good, then it transmits a RPTS packet, else does not transmit and goes for another back-off. $P_{RPTS} = P^i_t$, obeys the power constraint mechanisms as discussed above. The RPTS packet contains (P^i_t, P^i_{nS}) in it's packet, where P^i_{nS} is the source noise power. It also starts a timer $TIMER_{RPTS}$.
2. Node j receives a RPTS packet and measures the received power P^j_r. It computes the channel gain as $G^{ij} = \frac{P^j_r}{P^i_t}$. Then it computes the required transmitter power for node i as follows:

$$P^i_{t-des} = max\left(\frac{P^j_{r-des}}{G^{ij}}, \frac{\gamma_{des} P^j_{nD}}{G^{ij}}\right) \tag{3}$$

Where,P^j_{r-des} is the desired power level at node j for a correct reception and P^j_{nD} is the destination noise power measured at j, such that $P^j_{r-des} > P_{RX-th}$ and $\gamma_{des} > \gamma_{th}$, is satisfied. After computing the required transmitter power, the receiver j transmits an APTS packet to the transmitter node i.

$$P_{APTS} = P^j_t = max\left(\frac{P^j_{r-des}}{G^{ij}}, \frac{\gamma_{des} P^j_{nS}}{G^{ij}}\right) \tag{4}$$

An APTS packet contains (P^i_{t-des}) in its packet, such that the transmitter can transmit its data packets at this power level. We considered P^j_{nS} for computing P_{APTS} (Eqn. 4) and P^j_{nD} for P^i_{t-des} (Eqn. 3).
3. If node i receives the APTS packet before the expiry of $TIMER_{RPTS}$, then it checks for P^i_{t-des} from that packet and transmits the data packets at a power level of P^i_{t-des}, if, $P^i_{t-des} < P^i_{t-bound}$, else, goes to back-off mode. If it does not receive any APTS before the expiry of $TIMER_{RPTS}$, it goes to back-off.
4. When node j receives the data packet and starts transmitting the busy tone in the second channel at repeated interval. The duration of the busy tone pulse is based on the signal capture interval of the receiver. The busy tone power level is calculated as follows:

$$P^j_{t-BT} = \frac{P_{t-Max} P_{CS-th}}{E^j} \tag{5}$$

The above transmitting power level should be below the maximum busy tone transmitting power $P_{t-BT-Max}$. It also defines the minimum noise tolerance level E^j_{Min} as follows:

$$E^j_{Min} = \frac{P_{t-Max} P_{CS-th}}{P_{t-BT-Max}} \tag{6}$$

E^j_{Min} limits the ability of the busy tone to bound far away stations when the receiver is very sensitive to any small increase in noise. If there was not a minimum noise tolerance the busy tone power could potentially approach infinity and force nodes infinitely far away to not transmit at all. The advertised noise tolerance level E^j is found out as follows:

$$E^j = max\left(\frac{P^j_r}{\gamma_{th}} - P^j_{nD} E^j_{Min}\right) \tag{7}$$

5. When any potentially interfering node l receives this busy tone, it computes its transmission power bound as follows:

$$P^l_{t-bound} = \frac{E^j}{G^{jl}} \tag{8}$$

Similarly, it also receives busy tones from other nodes and hence can compute the effective transmit power bound as

$$P^l_{t-bound} = min\left[min_j\left(\frac{E^j}{G^{jl}}, P_{t-Max}\right)\right] \tag{9}$$

6. After receiving the entire data packet without errors, the receiving node j sends an ACK at the power lever needed to get back to the source on the data channel. $P^j_{ACK} = P^j_{t-des}$
7. After receiving a valid ACK, the transmitting node i it resets its back-off and returns to the $IDLE$ state otherwise, it increases the back-off and starts over again.

A.4 Performance of PCMA Protocol

Simulation studies [3] shows PCMA performs better as compared to IEEE Std 802.11 MAC protocol. This improves the throughput by a factor of two for densed network as compared to IEEE Std 802.11. Also, it increases the channel utilisation and hence the efficiency and preserve the collision avoidance property of multiple access protocols. But, it is silent over the following issues:
1. Mobility issues
2. Multi-hop wireless ad-hoc networks
3. Contention of busy tone
4. Fading of busy tone as compared to that of data

B. POWMAC

This is a single channel solution [4] for transmission power control in wireless ad-hoc networks. It is similar to IEEE Std. 802.11 ad-hoc MAC protocol in hardware sense, but differs in its implementation technique (software). POWMAC is equipped with only one trans-receiver and a carrier-sense hardware, similar to IEEE Std. 802.11. POWMAC is distributed, asynchronous and adaptive to channel changes. It does not transmit any control packets similar to RTS/CTS of IEEE Std. 802.11 to acquire a floor. Instead, it transmits its own form of control packet with and is used in conjunction with the received signal strength of these packets to dynamically bound the transmission power (TP) of potentially interfering nodes in the vicinity of a receiving node. POWMAC is based on the assumptions described in Section II-A.

B.1 Power Analysis of POWMAC

Similar to PCMA, it defines a minimum SINR (γ_{th}) for a target bit-error-rate at the receiver. For, any successful transmission between the nodes i and j, the condition $\gamma^j \geq \gamma_{th}$ must holds good. It defines another term P^j_{MAI} as the multi access interference (MAI) from interfering transmissions at receiver j. This is related to the $noiserise$ or $loadfactor$ (ϵ^j) of the receiver as follows.

$$\epsilon^j = \frac{P^j_{MAI} + \eta_0}{\eta_0} \tag{10}$$

So, the minimum transmit power P^i_{t-Min} required at node i can be expressed as

$$P^i_{t-Min} = \frac{\gamma_{th}\epsilon^j\eta_0}{G^{ij}} \tag{11}$$

POWMAC finds out the noise tolerance at the maximum load (assuming the attenuation in the channel is by $(\frac{k}{d^n})$ (where, k is a constant, d is the distance and n is the loss factor, such that, $n \geq 2$)) ϵ_{Max} as follows:

$$\epsilon_{Max} = \frac{kP^{ij}_{load}}{\gamma_{th}\eta_0 d^n} \tag{12}$$

Where P^{ij}_{load} is the TP at maximum load needed by i to reach j. The maximum load is a design parameter in POWMAC.

B.2 POWMAC Protocol Steps

Similar to PCMA, we assume node i and j are communicating nodes and node i wants to start data communication with node j. Node k and l are potentially interfering neighboring nodes. Initially all nodes are in *IDLE* state and maintain a Power Constraint List (PCL_i). This list is an extension of Network Allocation Vector (NAV) [1] of IEEE Std. 802.11. This list (PCL_j) at node j contains the following entries in it.

- Address of each active node u in the vicinity of j.
- The path gain G^{ju} between the node u and j.
- Maximum tolerable interference P^u_{MTI} of node u. This is the maximum additional interference that any node u can tolerate from node j such that u's reception is not disturbed.
- Start time and duration of u's activity.

The steps of the POWMAC protocol are described below.

1. Node i finds out the maximum allowable transmit power P^i_{MAP}, such that it will not disturb its neighbors as follows:

$$P^i_{MAP} = min_{u \in PCL^i}\left[min\left(\frac{P^u_{MTI}}{G^{iu}}, P_{t-Max}\right)\right] \tag{13}$$

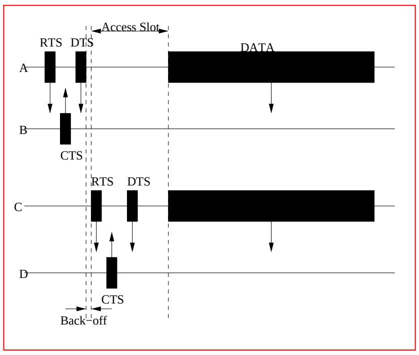

Fig. 4. Basic operations of POWMAC

Then, it (node i) transmits an RTS packet with $P_{RTS} = P_{t-Max}$. The RTS packet contains the following items: $(i, j, P^i_{MAP}, N^i_{AW}, T^{ij}_{data})$. Where, N^i_{AW} is the remaining number of the access slots in the current access window (AW) and T^{ij}_{data} is the transmission duration of i's data packet. AW allows other neighboring nodes to exchange their control packets such that (interfering) data transmissions can proceed simultaneously (as long as collisions are prevented). AW is needed for two reasons. They are: (i) it reduces the likelihood of collisions between control and data packets and (ii) the transmitting nodes can inform other nodes that are currently transmitting or receiving of the ensuring data transmission. The basic operation of POWMAC is described in Figure 4.

2. Node j receives the RTS packet and computes the path gain G^{ij} as before. Now it calculates the minimum transmission power of node i for its data packets such that it can correctly decode them. This is computed as follows:

$$P^{ij}_{t-Min} = \frac{\gamma_{th}[\eta_0 + P^j_{MAI-cur}]}{G^{ij}} \tag{14}$$

Where,$P^j_{MAI-cur}$ is the current MAI from all already ongoing (interfering) transmissions. Node j transmits a valid CTS packet if the following conditions are satisfied: $P^i_{MAP} \geq P^{ij}_{load} \geq P^{ij}_{t-Min}$, else it transmits a negative CTS packet. Also, node j updates its $P^j_{MAI-future}$ as follows:

$$P^j_{MAI-future} = \frac{G^{ij}}{\gamma_{th}}\left[P^{ij}_{load} - P^{ij}_{t-Min}\right] \tag{15}$$

This is approximated as

$$\begin{aligned} P^j_{MAI-future} &= P^j_{MAI-within} + P^j_{MAI-other} \\ &= (1+\alpha)P^j_{MAI-within} \end{aligned} \tag{16}$$

Where, $P^j_{MAI-within}$ is the MAI at node j due to the interference that are within the maximum range of node j and $P^j_{MAI-other}$ is the MAI at node j due to the interference outside of the maximum range of node$j$. The value of α is found to be 0.5 (from 2-ray model [4]). So, P^j_{MTI} can be computed as:

$$P_{MTI}^{j} = \frac{P_{MAI-future}^{j}}{(1+\alpha)N_{AW}^{j}} \quad (17)$$

Computation of P_{CTS}^{j} is done as follows:

$$P_{CTS}^{j} = min\left(\gamma_{th}\eta_0 \frac{P_{t-Max}}{P_{MTI}^{j}}, P_{t-Max}\right) \quad (18)$$

The CTS packet consists of:
$(j, i, P_{load}^{ij}, P_{MTI}^{j}, P_{CTS}^{j}, N_{AW}^{j}, T_{data}^{ij})$
3. After receiving the CTS packet from node j, node i transmits another control packet called Data-To-Transmit (DTS) to node j in the control channel. The aim of this packet is to inform the i's neighbor that may not have heard the CTS and RTS. It is similar to carrier sensing [1] of IEEE Std. 802.11. The power level of DTS packet is such that, $P_{DTS} = P_{t-Max}$ and it contains (j, i, P_{load}^{ij})
4. If, any node l is a potential interfering node, then, after receiving the RTS packets from all transmitting nodes, it updates its P_{MAP}^{l} (similar to Eqn. 13), such that, it will not create any problem to the communication between i and j.
5. When another node s receives i's DTS packet, it computes the path gain G^{si} and the interference level at s due to i's data transmission to j. This information is used to figure out whether or not, it can receive any data packet following the current AW.
6. The value of AW is decremented by 1, if $I^i < 0.75(\epsilon_{max}\eta_0)$, else, remains unchanged, where, I^i is the total interference perceived at node i. The AW size is incremented or decremented by a step of 1 always to avoid large fluctuations.
7. After transmitting a DTS packet, node i waits till the end of the current access window and transmits the data packet at $P_{data}^{i} = P_{load}^{ij}$.
8. After a successful reception of a data packet at node j, node j responds with an ACK packet and is transmitted in the data channel at the same power level that would have been used if node j were to be send a data packet to node i. Enough Forwarding Error Correcting (FEC) code is used to protect the ACK packet.

B.3 Performance of POWMAC

POWMAC is the first protocol to use single channel power control mechanism for non-CDMA based wireless ad-hoc networks. Though, it does not consider mobility, but this model holds good for mobility as the coherent time due to mobility is more than the data transmission window. Also, it effectively utilises the wireless channel and hence achieves better throughput. It achieves more than 45 percent in throughput as compared to IEEE Std 802.11.

C. DCA-PC

Dynamic Channel Assignment with Power Control (DCA-PC) [5] protocol is very much similar with PCMA. This power control protocol uses one *control channel* to transmit all the control packets (RTS, CTS, RES etc) where as it uses multiple *data channels* for data transmission, which are assigned on demand. In DCA-PC, a pair of source and destination nodes uses a RTS/CTS dialogue to decide which channel to reserve and which power level to use for data transmission. A RES message is used to reserve a data channel. The data and ACK packets are transmitted on the reserved data channel using the required power level. This an *on-demand* method of assigning channels to the mobile nodes. The number of channel is independent of the network topology and degree. DCA-PC is actually an extension of another MAC protocol called *Dynamic Channel Assignment (DCA)* [9]. DCA uses single fixed power technique, where as DCA-PC is a power controlled technique.

C.1 Analysis of DCA-PC Protocol

DCA-PC [5] protocol resolves the problems of channel assignment (based on DCA protocol scheme), medium access and power control in an integrated manner. As discussed above, the transmitting node i reserves a data channel whenever it desires to transmit data, through an RTS/CTS dialogue and releases the channel after the transmission is over. The data channels $D_1, D_2, ..., D_N$ are distinct non-overlaping frequency bands in a given bandwidth. The control channel is also a narrow band channel in the given bandwidth and is distinct from the data channels. Each node is equipped with two half-duplex trans-receivers *control trans-receiver* and *data trans-receiver.* Each node i maintains an array called *POWER[..]* for each neighboring node j. This array carries the level of power that node i should be used to reach any neighboring node j. These values are dynamically adjusted. If, any node k, which was a neighbor before, moved out, then the entries for k in the *POWER[..]* array is updated to ∞. The control packets are transmitted at a maximum power P_{t-Max}. Similarly, P_{RX-th}^{j} is the minimum power level required at a receiving node j for error free data receiving. When node i does not hear anything from its neighbor j in a specified time (set by a timer), it updates the $POWER[j] = \infty$. Each node maintains another data structure called *channel usage list (CUL[..]).* At any node i, $CUL[j]$ keeps the records of any neighboring node j as follows:

1. $CUL[j].host$ - neighbor of node i
2. $CUL[j].ch$ - data channel used by node j
3. $CUL[j].rel-time$ - releasing time of channel $CUL[j].ch$ by node j

Each node i also maintains another variable called *free channel list (FCL).* This is also updated dynamically.

C.2 DCA-PC Protocol Steps

As discussed previously, node i desires to transmit data packets to node j. For this, DCA-PC protocol steps are as follows:

1. Before transmission from node i to node j, DCA-PC protocol first checks the following two conditions.
- Node j is not equal to any $CULl.host$, such that

Variables	*Meanings*
T_{SIFS}	Length of short inter-frame spacing
T_{DJFS}	Length of distributed inter-frame spacing
T_{RTS}	Time to transmit an RTS
T_{CTS}	Time to transmit a CTS
T_{RES}	Time to transmit a RES
T_{CURR}	The current clock of a mobile host
T_{ACK}	Time to transmit an ACK
NAV_{RTS}	Net. alloc. vector on receiving an RTS
NAV_{CTS}	Net. alloc. vector on receiving a CTS
NAV_{RES}	Net. alloc. vector on receiving a RES
L_d	Length of a data packet
L_c	Length of a control packets
B_d	Bandwidth of a data channel
B_c	Bandwidth of the control channel
τ	Maximal propagation delay

TABLE I
VARIABLES AND CONSTANTS USED IN DCA-PC

$$CUL[l].rel-time > T_{CURR}+(T_{DIFS}+T_{RTS}+T_{SIFS}+T_{CTS}) \tag{19}$$

If so, node j is $BUSY$ after a successful exchange of RTS and CTS packets.

- There must be a channel D_k, such that for all node l:

$$(CUL[l].ch = D_k) \implies (CUL[l].rel - time \leq T_{CURR} + (T_{DIFS} + T_{RTS} + T_{SIFS} + T_{CTS})) \tag{20}$$

This ensures that the channel D_k is either not present in the CUL or is present in CUL but will be free after a successful exchange of RTS and CTS packets.
If the above two conditions are satisfied, node i updates FCL by adding all D_ks which satisfies the conditions.
2. Similar to IEEE Std. 802.11 node i senses the channel for T_{DIFFS} duration and if it is free, then transmits an RTS packet to node j at a power level of $P_{RTS} = P_{t-Max}$. It contains (FCL, L_d). Else, it goes to back to step 1.
3. Node j receives the RTS packet. Then it checks whether any data channel D_k belongs to the received FCL (through RTS packet) such that $\forall l$:

$$(CUL[l].ch = D_k) \implies ((CUL[l].rel - time) \leq (T_{CURR} + T_{SIFS} + T_{CTS}) \tag{21}$$

If true, the D_k is a free channel and can be used. Then node j selects D_k from the list and replies with a CTS packet to node i at a power level of $P_{CTS} = P_{t-Max}$. The CTS packet contains (D_k, NAV_{CTS}). The value of $NAV_{CTS} = \frac{L_d}{B_d} + T_{ACK} + 2\tau$. Further node j tunes its data trans-receiver to the channel D_k.
Else, node j replies with a $CTS(T_{est})$ packet to node i, where, T_{est} is defined as

$$T_{est} = min\Big(\forall l, CUL[l].rel - time\Big) - \Big(T_{CURR} + T_{SIFS} + T_{CTS}\Big). \tag{22}$$

4. When any potentially interfering node s receives the RTS packet sent by node i to node j, waits for NAV_{RTS} duration to avoid s from interrupting the RTS/CTS dialogue between i and j, before using the channel. Where NAV_{RTS} is defined as:

$$NAV_{RTS} = 2T_{SIFS} + T_{CTS} + T_{RES} + 2\tau. \tag{23}$$

5. If node i does not receive any CTS packet within a time period of $T_{SIFS} + T_{CTS} + T_{RES} + 2\tau$, after sending the RTS packet to node j, it retries again until reaching the maximum number of retries.
6. When node i receives node j's CTS packet, it first updates the CUL entries as: $CUL[l].host = j, CUL[l].ch = D_k$ and $CUL[l].rel - time = T_{CURR} + NAV_{CTS}$. Then it broadcasts the RES packet on the control channel at P_{t-Max} power level. It contains (D_k, NAV_{RES}), where,

$$NAV_{RES} = NAV_{CTS} - T_{SIFS} - T_{RES}. \tag{24}$$

7. Then node i starts sending the data packets to node j on the data channel D_k. Sending of data packets and RES packets starts at same time.
8. If node i receives $CTS(T_{est})$ packet, then it starts from step 1 after a duration of $T_{CURR} + T_{est}$ or when a data channel is released.
9. When any potentially interfering node s receives the CTS packet sent by node j, it updates its CUL as: $CUL[l].host = j, CUL[l].ch = D_k$ and $CUL[l].rel - time = T_{CURR} + NAV_{CTS} + \tau$. If, it receives $CTS(T_{est})$ packet, it ignores.
10. When node s receives a RES (D_k, NAV_{RES}) packet, it updates its CUL as: $CUL[l].host = i, CUL[l].ch = D_k$ and $CUL[l].rel - time = T_{CURR} + NAV_{RES}$.
11. When node j receives the data packets completely, it transmits an ACK packet in data channel D_k at a power level of P_{t-Max}.

C.3 Performance of DCA-PC

DCA-PC is the first protocol to realize the mechanisms of power control and multi channel medium access together in MAC protocols of wireless ad-hoc networks. By using multiple channels, it is easier to increase the throughput, reduce normalized propagation delay per channel, and support quality of service. The basic problem in this protocol is that it does not perform well at high load and high data rate. Also, there are some interference due to non orthogonality of data channels.

IV. CDMA based Power Control Mechanism

In CDMA based wireless ad-hoc network, nodes serve as CDMA trans-receivers. The advantage of CDMA [10] in ad-hoc networks are mainly: (i) capacity improvement, (ii) energy saving, and (iii) routing overhead reduction. For CDMA based systems, the SINR expression [8] in Eqn. 1 is modified as:

$$\gamma^i = \left(\frac{W}{R}\right)\left(\frac{G^{ij}P_t^j}{\sum_{j=1,j\neq i}^{N} G^{ij}P_t^j + \eta^i}\right) \quad (25)$$

Where, W is the spreading bandwidth and R is the data rate. The ratio of W/R is equal to unity in non-spreading case (Eg. PCMA), where as it is greater than unity in spreading case.

A. IBMA

IBMA [6] is basically the CDMA version of PCMA protocol. Similar to PCMA, IBMA uses two distinct non-overlapping channels and are known as *Controlchannel* and *datachannel*. The control channel is a narrow band channel and is used for sending the control packets, where as the data channel is a wide-band channel and is used for RTS, CTS, data and ACK packets.

The control packets are transmitted without any spreading, where as the packets in the data channel are transmitted with spreading. Further, it defines three different kinds of control packets and are as follows: (i)*Beacons*, (ii) *Data-Send (DS)* and (iii) *Interference Tolerance (IT)*. Beacons are used to broadcast the noise tolerance of the nodes and to calculate the path gain between nodes. DS packets are used to educate the transmitter about the data transmission power level, where as IT packets are used to educate the other nodes about the excess tolerance level that the node can tolerate.

This protocol assumes all nodes are static and are assigned with unique spreading codes. The spreading is done by the use of non-orthogonal codes. The data channel receiver is tuned to it's own spreading code and data transmission and hand shaking are done with the receiver's spreading code.

Each node in IBMA protocol maintains a *Neighborhood Table*, which contains information about its first-hop neighbors. The neighborhood table of node i contains $(j, spreading-code^j, status^j, IT^j, G^{ij})$, in it for each neighboring node j. In IBMA, the path gain in data and control channel are different, but all the assumptions described in Section II-A holds good.

A.1 IBMA Protocol Steps

1. During *IDLE* state, each node i broadcasts its beacons in the control channel. It contains $(i, spreading-code^i, IT^i)$. It is transmitted at a fixed power level P_{beacon}. On receiving this, any node j computes its path gain for control and data channel.

$$G_c^{ij} = \frac{P_r^j}{P_{beacon}} \quad (26)$$

$$G^{ij} = G_c^{ij}\left(\frac{\lambda_d}{\lambda_c}\right)^2 \quad (27)$$

where, G_c^{ij} is the path gain for control channel and G^{ij} is path gain for data channel [8], [10].

2. When any node i desires to transmit data to any node j, it first checks the neighborhood table to know j's status. If that is free, then it finds out the feasibility of sending data to that as follows.

$$P^i_{t-Minreq} = \left(\frac{R}{W}\right)\gamma_{th}\left(\frac{IT^j}{G^{ij}}\right) \quad (28)$$

This value is an approximate value and depends on the staleness of information in the power map. Similarly, node i computes the maximum permissible power map $P^i_{t-Maxper}$, which can start without disrupting other on-going transmission.

$$P^i_{t-Maxper} = min\left[\left(min_{j\in table}(IT^j)\right), P_{t-Max}\right] \quad (29)$$

If $P^i_{t-Minreq} > P^i_{t-Maxper}$, node i backs off, else, sends an RTS packet.

$$P^i_{RTS} = \frac{P^i_{t-Minreq} + P^i_{t-Maxper}}{2} \quad (30)$$

The RTS packet contains (i, P^i_{RTS}, IT^i).

3. On receiving RTS from node i, node j calculates the path gain G^{ij} and computes the minimum power at which node i should send its data packets. Similar to Eqns. 28 and 29 it computes both $P^i_{t-Minreq}$ and $P^i_{t-Maxper}$ values. Further, it computes the P^i_{data} as follows:

$$P^i_{data} = \frac{P^i_{t-Minreq} + P^i_{t-Maxper}}{2} \quad (31)$$

Node j communicates this value to node i through a CTS packet. The transmission power of CTS packets are calculated similar to that of RTS packets in Eqns. 28, 29 and 31.

$$P^j_{CTS} = \frac{P^j_{t-Minreq} + P^j_{t-Maxper}}{2} \quad (32)$$

The CTS packet contains (j, P^i_{data}) in its packet.

4. After receiving the CTS packet from the node j, node i transmits a DS packet in the control channel with $P^i_{DTS} = P^i_{data}$, such that all nodes update their IT values as per the commencing transmissions. The DS packet contains (i, P^i_{data})

5. On receiving a DS packets of node i, any node j updates its path gain and updates its entries in the neighborhood table with the status of the node as *TRANSMITTING*. Since the length of data packets are constant, all nodes calculates the exact duration of data transmission and increase their IT values after this duration.

6. After receiving DS packets, the receiving node j, advertises its excess tolerance level through IT packets. It computes its excess interference tolerance as follows:

$$IT^j = \left(\frac{W}{R}\right)\left(\frac{P^j_{r-data}}{\gamma_{th}}\right) - IT^j_{current} \tag{33}$$

Node j transmits IT packets and contains (IT^j) in it.
7. On receiving an IT packet, any node u updates its current interference tolerance value in its neighborhood table.
8. The transmitting node i also computes its interference tolerance value and updates its neighborhood table with the new entries of IT^j and changed the status flag $status^j$ as *RECEIVING*.
9. IT packets are sent whenever the noise tolerance at a node changes, while the beacons are sent periodically.
10. All the control packets are used by the single-hop neighbors to update their neighborhood tables.
11. When node j receives data successfully, it transmits an ACK packet. P^j_{ACK} is calculated in the same manner in which P^j_{CTS} was calculated.
12. If node i does not receive any ACK packets in a stipulated time (set by a timer), it times out and goes back-off.

A.2 Performance of IBMA Protocol

IBMA protocol performs better as compared to PCMA protocol. At both low data rate and higher data rate IBMA perform better than PCMA. Since multiple parallel transmission is possible in IBMA, the contention in IBMA is lesser as compared to PCMA. Hence, it performs well and results higher throughput than PCMA. Another significant achievement of IBMA is the delay jitter. The delay jitter in IBMA is very less as compared to that of PCMA. This is because, as the load increases the contention resolution in PCMA results in large back-offs, where as the multiple parallel transmission in IBMA results significance improvement in back-offs and hence smaller delay jitter.

B. CA-CDMA

Controlled Access CDMA (CA-CDMA) [7] protocol is proposed to solve the near-far problem of CDMA ad-hoc networks. Similar to IBMA, it is also a two channel solution. The two distinct non-overlapping channels and are known as *control channel* and *data channel*. The control channel is a narrow band channel and is used for sending the control packets, where as the data channel is a wide-band channel and is used for data and ACK packets. The spreading code in the control channel is common for all nodes and are used to contend for the channel resource with a modified RTS/CTS mechanism. The data packets are transmitted in the data channel encrypted with different node-specific spreading codes. The distribution of spreading codes can be done according to any code assignment scheme. CA-CDMA claims that, even if the code assignments are not proper, CA-CDMA still functions properly. Similar to IBMA, multiple parallel transmission is also possible in CA-CDMA. This protocol controls *multi-access-interference* (MAI) by dynamically adjusting the transmission power during data transmission. It also assumes that there is no interference between the data and control channel. Along with this the other assumptions described in Section II-A holds good. This is the first protocol to address the near-far problem of CDMA based wireless ad-hoc networks.

B.1 Power Analysis of CA-CDMA

The power analysis of CA-CDMA is similar to that of POWMAC, though POWMAC is a single channel solution and a non CDMA-based MAC protocol. Similar to Eqn. 25, CA-CDMA defines the SINR at any node i as γ^i. Like PCMA, the path gain in control channel and data channel is CA-CDMA is assumed to be same. In CA-CDMA, the choice of the power level is a trade-off between link quality and MAI. Similar to POWMAC, it also computes the interference margin. The Equations 10 11 and 12 of POWMAC hold good in CA-CDMA, where $P^i_{MAI} = \frac{2\sum_{j=1,j\neq i}^{N} P^j}{3W}$ (for an asynchronous direct sequence BPSK system [7]) is the total MAI at any node i.

B.2 CA-CDMA Protocol Steps

As discussed before, node i desires to transmit data to node j and node k, l are potentially interfering nodes placed in the neighborhood of i and j. The protocol are explained below.
1. Node i transmits an RTS packet with $P^i_{RTS} = P_{t-Max}$ in the control channel. The RTS packet contains $(i, j, P^i_{MAP}, T^{ij}_{data}, FrameControl, FCS)$ in it. P^i_{MAP} is the maximum allowable power level that node i can use without disturbing any ongoing reception. P^i_{MAP} is computed in a similar manner to that of POWMAC (Eqn. 13).
2. Node j receives the RTS packet from node i in the control channel. It computes the path gain similar to that of POWMAC. As described in Eqn. 14, it computes the minimum transmission power at which node i needs to transmit its data packets for a successful transmission. Also, it computes the maximum allowable transmission power for node i as follows:

$$P^{ij}_{allowed} = \frac{\epsilon_{Max}\gamma_{th}\eta_0}{G^{ij}} \tag{34}$$

$P^{ij}_{allowed}$ of CA-CDMA is similar to P^{ij}_{load} of POWMAC. Node j transmits a valid CTS packet if the following conditions are satisfied: $P^i_{MAP} \geq P^{ij}_{allowed} \geq P^{ij}_{t-Min}$, else it transmits a negative CTS packet. Also, it calculates the interference tolerance P^j_{MTI} (similar to Eqn. 17). The CTS packet consists of $(j, i, P^{ij}_{allowed}, P^j_{MTI}, T^{ij}_{data}, FrameControl, FCS)$ The power level of CTS packet $P^j_{CTS} = P_{t-Max}$.
3. After receiving the CTS packet, node i transmits the data packets at a power level of $P^{ij}_{allowed}$. If, it does not receive any CTS packet in a stipulated time period, it goes to back-off mode and restarts again.
4. When a potentially interfering node l receives a CTS packet, it computes the path gain G^{jl}, and uses P^j_{MTI} level of the CTS packet to compute P^l_{MAP} (similar to Eqn. 13).
5. After a successful reception of data packet at node j, node j responds with an ACK packet and is transmitted in the data channel at the same power level that would have

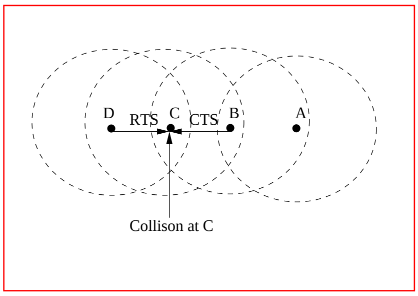

Fig. 5. Collision due to large duration of control packets

been used if node j were to be send a data packet to node i. Enough FEC code is used to protect the ACK packet.

B.3 CA-CDMA Performance

As explained in [11], when the transmission and propagation times of control packets are long, there is a strong possibility of collision between a CTS packet and an RTS packet of another contending node. This is explained in Figure 5. Suppose node D starts sending an RTS packet to node C while C is receiving B's CTS packet which is intended to the node A. A collision occurs at C, and hence, C is unaware of B's subsequent data reception. Afterwords, if C decides to transmit a CTS to D, and may destroy B's reception. At high loads, such a collision can lead to collisions with data packets.

CA-CDMA avoid this problem by the use of $P_{MAP}^{(.)}$ and $P_{allowed}^{(.)}$. When $P_{MAP}^{i} \leq P_{allowed}^{ij}$ becomes true a special CTS packet is being sent by the receiving node j over the control channel preventing the RTS sender from commencing its new transmission. The duration field of the special CTS packet contains the time left for node i to finish its ongoing reception. In this manner it avoids collision due to large propagation times of control packets. CA-CDMA relies upon the other layer, if any other collision occurs.

So far as performance is concerned, CA-CDMA achieves up to 280 percent increase over the throughput of the IEEE Std 802.11 scheme. This increase is attributed to the increase in the number of parallel transmissions. Further, CA-CDMA saturates at about twice the load at which the IEEE Std. 802.11 scheme saturates and hence is a good protocol as compared to IEEE Std 802.11 at any load. The results also show that CA-CDMA requires less than 50 percent of the energy required as compared to the IEEE Std 802.11 scheme. Energy savings in CDMA is due to the use of orthogonal codes resulting channel reuse.

V. Conclusion

We have discussed both CDMA based and non-CDMA based MAC protocols in our report. From different protocols, we have learnt that transmission power control can significantly increase the throughput and battery life. We have not discussed the implementation issues of the above discussed protocols. Also, we have not discussed (a) capacity optimisation, (b) support for real-time traffic (QoS based), (c) multi-class and multi-hop communication, (d) load vs. information rate balancing etc., in in our report. These issues need to be looked upon during the implementation stages.

Acknowledgments

I would like to express my deep sense of gratitude to **Prof. Abhay Karandikar** for his motivation and fresh ideas he provides during the course of my seminar.

References

[1] LAN/MAN Standards Committee, *IEEE Standards for Local and Metropolitan Area Network:Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications.* IEEE Computer Society, 1999.
[2] J. H. Schiller, *Mobile Communications.* Addison-Wesley, 2000.
[3] J. Monks, V. Bharghavan, and W. Hwu, "A power controlled multiple access protocol for wireless packet networks," in *Proc. of IEEE INFOCOM, 2001*, pp. 219–228, 2001.
[4] A. Muqattash and M. Krunz, "A single-channel solution for transmission power control in wireless ad hoc networks," in *MobiHoc'04*, pp. 210–222, May 2004.
[5] Y.-C. Tseng, C.-Y. Lin, and J.-P. Sheu, "A multi-channel MAC protocol with power control for multi-hop mobile ad hoc networks," *The Computer Journal (SCI)*, vol. 45, no. 1, pp. 101–110, 2002.
[6] M. Singh, P. Sahni, J. Vyas, K. Paul, and A. Karandikar, "Interference based multiple access (ibma) protocol for cdma based ad hoc networks," in *Proc. of GLOBECOM, 2003*, 2003.
[7] A. Muqattash and M. Krunz, "Cdma-based mac protocol for wireless ad hoc networks," in *MobiHoc'03*, pp. 153–162, June 2003.
[8] A. J. Viterbi, *CDMA: Principles of Spread Spectrum Communication.* Addison-Wesley Wireless Communication Series, 2000.
[9] S.-L. Wu, C.-Y. Lin, Y.-C. Tseng, and J.-P. Sheu, "A new multichannel MAC protocol with on-demand channel assignment for multi-hop mobile ad hoc networks," in *International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN'00)*, pp. 232–237, December 2000.
[10] J. S. Lee and L. E. Miller, *CDMA Systems Engineering Handbook.* Artech House, 2002.
[11] Z. J. Haas and J. Deng, "Dual Busy Tone (DBTMA) - A Multiple Access Control Scheme for ad hoc Networks," *IEEE Transactions on Communications*, vol. 50, pp. 975–985, June 2002.