Incentive Compatible Medium Access Control in Wireless Networks*

Nassir BenAmmar
Institute for Systems Research
Dept. of Electrical and Computer Engineering
University of Maryland
College Park, MD 20742
nassir@isr.umd.edu

John S. Baras
Institute for Systems Research
Dept. of Electrical and Computer Engineering
University of Maryland
College Park, MD 20742
baras@isr.umd.edu

ABSTRACT

The current IEEE 802.11 medium access control standard is being deployed in coffee shops, in airports and even across major cities. The terminals accessing these wi-fi access points do not belong to the same entity, as in corporate networks, but are usually individually owned and operated. Entities sharing these network resources have no incentive in following protocol rules other than to optimize their overall utility, usually a function of throughput and delay. We briefly discuss shortfalls of the current IEEE 802.11 standard in environments where terminals are competing for a common bandwidth resource, and then we introduce a new MAC protocol designed with the above considerations. Thus the new Incentive Compatible MAC (ICMAC) protocol is more suited for these open environments, without compromising the overall network performance.

Keywords

Wireless Networks, Medium Access Control, Vickrey Auction

1. INTRODUCTION

Most network protocols today are designed with the objective of maximizing performance of the network with respect to a set of network criteria, typically a function of throughput and delay, with the assumption that all participating entities of the network will follow protocol rules. This assumption has not been a major issue in wired networks due to the reliable medium and the abundance of bandwidth. However, this is not the case in wireless networks due to the broadcast nature of the wireless medium and the stringent bandwidth limitations. It has been shown that the de-facto medium access control for wireless networks, in particular the IEEE 802.11 protocol, suffers from many security weaknesses. A lot of work has been done to improve this MAC protocol. Security issues were of various types. Some involved the mechanism of association and authentication; others were at the message encryption protocol. However, the focus of this paper is on the inherent access control mechanism. Various access techniques have been used in multiuser communications allowing communicating entities to share common bandwidth. Time division multiple access divides the time axis into time slots and assigns individual slots to various users in a round robin fashion. Similarly, frequency division multiple access divides the frequency domain into channels used by various terminals. Both these access schemes are not appropriate for data traffic as traffic is of a bursty nature and results in wasted resources when users are assigned slots but have no traffic to send. Code division multiple access and frequency division multiple access take advantage of both frequency and time domain by means of spreading codes allowing concurrent multiple transmissions. Complexity of these systems resides at the access point (AP). The multi-user communication techniques described above are broadly used in cellular networks, where the network is designed to sustain a given number of users at any given time. Another widely used multiuser access mechanism for wireless data networks is random multiple access. The simplest form of random access is ALOHA, where a node access the channel if it has a data packet to transmit and waits a random number of slots if it experiences a collision. Progressively more techniques and improvements have been added to prevent collision at the access channel. MACA, MACAW and IEEE 802.11 are examples of protocols incorporating some of these collision avoidance techniques. Physical carrier sensing, virtual carrier sensing and exponential backoff timer are all used in IEEE 802.11 distributed coordination function (DCF) in order to reduce collision rate and get a better network throughput [1]. Due to the random nature of channel access, stations have an incentive to deviate from protocol rules by altering transmission and backoff probabilities to gain better performance. Previous papers have addressed the noncooperative behavior in a random access MAC. [14] studied the stability region of a slotted ALOHA system with selfish users for a general multipacket reception model. The model assumes a perfect information on

*This material is based upon work partially supported by the U.S. Army Research Office under Award No DAAD 190110494 and partially prepared through collaborative participation in the Communications and Networks Consortium sponsored by the U. S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011.

GameNets'06, October 14, 2006, Pisa, Italy

the number of contending nodes. The authors show that the stability region is a function of the station transmission cost. Unlike [14], [10, 9, 2] consider a finite station ALOHA system. [10, 9] assume that n heterogeneous stations are always backlogged and the network charges M for successful transmission. Sequentially, each user broadcasts its transmission probability and solves a utility maximization problem using the expected throughput (from other stations advertised transmission probabilities) and the network transmission charge. The network adjusts the charge so as to achieve a target throughput. However, this system seems unstable due to the inelasticity of bandwidth requirements as users demands are switched on and off when the network price oscillates around their willingness to pay. [2] considers both a cooperative team problem and a noncooperative game problem formulation. The differences of throughput and transmission probability solution of both models as a function of node arrival probabilities are highlighted. The authors also point to the transmission cost as the deteriorating factor of throughput in the noncooperative game. Using an extension of [4], [11] analyzes station performance as the number of selfish users/stations increases. An extreme selfish strategy is considered and a collective punishment strategy in the case of selfish behavior detection to achieve Nash equilibrium for a liminf-type asymptotic utility. The rest of the paper is organized as follows. In section 2, we summarize the operation of the distributed coordination function of IEEE 802.11 and its vulnerability at the access channel. In section 3, we use game theory to explain the emergent behavior of rational entities in a random access channel and its effect on throughput. The findings naturally lead to an auction mechanism to alleviate some of the problems associated with the random access. Then we introduce a new Incentive Compatible Medium Access Control scheme in section 4 and discuss performance and design parameters in 5. We finally show simulation results pertaining to design and performance issues.

2. THE DISTRIBUTED COORDINATION FUNCTION OF IEEE 802.11

The distributed Coordination Function (DCF) has two access modes, the RTS/CTS mode and the basic mode. In the RTC/CTS mode, a node with a packet to transmit first senses the medium and if found idle picks a random waiting time before it reserves the wireless medium. The medium reservation is done by the exchange of a Request to Send (RTS) and Clear to Send (CTS) messages. With this exchange of messages, the other nodes are notified that the medium will be busy for a duration advertised in the RTS packet and then updated in the CTS packets. Thus terminals in the vicinity of the transmitter as well as those in the vicinity of the receiver are aware of the transmission (assuming these messages are detected correctly) and update their Network Allocation Vector (NAV). NAV informs a node about an ongoing transmission without continuously sensing the medium. This is referred to as virtual transmission sensing as opposed to physical transmission sensing. For instance, the duration advertised in RTS consists of the time required to transmit the data frame, plus the CTS frame, plus the ACK frame, plus three SIFS intervals. The SIFS interval is the short interframe interval required between the RTS, DATA, CTS, and ACK frame. In the basic mode, a node starts transmitting its data traffic after a random waiting time without the exchange of the RTS and CTS control packets.

2.1 Exponential Backoff Mechanism

During a transmission, a collision can occur for various reasons. It can happen if 2 nodes attempt to transmit at the same time, or if one node does not detect neither RTS nor CTS packet belonging to the upcoming data transmission and attempts to transmit while another data transmission is ongoing. Also a loss of an RTS or CTS packet can be considered as a collision by the initiating transmitter. The collision detection is unlike that of wired medium access as nodes are not capable of transmitting and receiving at the same time. In addition to the physical and virtual carrier sensing, an exponential backoff mechanism is in place to reduce collision rate. Before transmitting, each node picks a random waiting time from a uniform distribution between 0 and $CW - 1$. CW is the contention window size and it follows an exponential increase with the number of experienced collisions up to a maximum CW_{max}.

$$CW = \begin{cases} 2^i CW_{min} & \text{if } i < m \\ 2^m CW_{min} = CW_{max} & \text{if } i \geq m. \end{cases} \tag{1}$$

Here CW_{min} is the starting window size and i is the number of collisions experienced by the packet. Upon successful transmission, the window size CW gets reset to CW_{min}. The random backoff selected corresponds to the number of slots a station needs to wait before attempting to transmit. The backoff timer is decremented only when the medium is idle; when the medium becomes busy the backoff timer freezes and resumes once the current transmission finishes. Fig. 1 illustrates this mechanism. In this case when A and D transmit to B, C freezes its backoff counter.

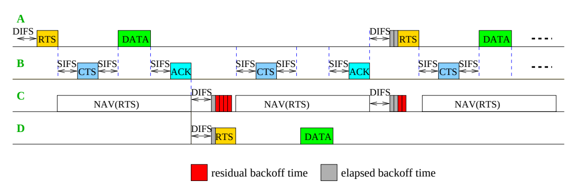


Figure 1: IEEE 802.11 Backoff Operation

2.2 Shortfalls of the Random Backoff Time

The protocol was designed for networks where all the entities participating obey the protocol rules. This assumption is valid if the network is owned by the same entity. For example, company networks, rescue and relief mission networks. However this will not apply in a network where nodes are individually owned and controlled, and are competing for the same network resources. There are many existing networks of this form and more are being deployed. These networks are being deployed in major cities, coffee shops, airports,etc... Some are provided free of charge or as complementary service, with an espresso for instance, others charge users according to time of use, in some airports for example, whether or not traffic is sent. Before we proceed further we divide users into three categories from a security standpoint.

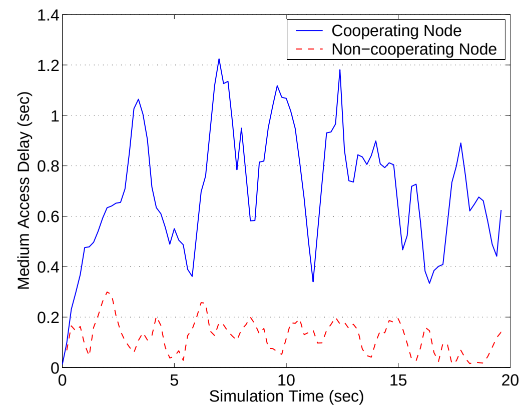
1. Well behaved user: This refers to a user/station obeying the exact rules of the protocol.

2. Selfish user: This refers to a user that might not follow exact protocol rules in order to gain more bandwidth, shorter delay, and a better overall performance.

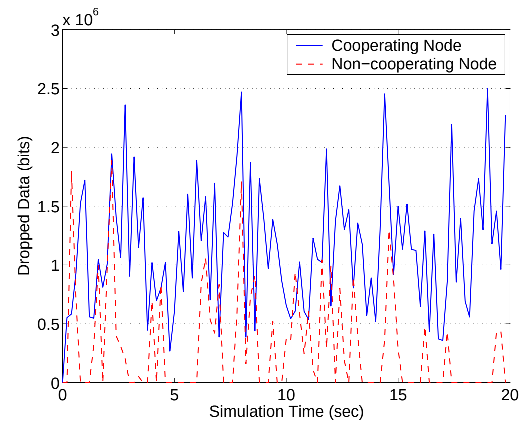
3. Malicious user: This refer to a user that has an objective of disrupting the network operation.

A selfish station might choose a short backoff time after a collision instead of choosing a random backoff time from the uniform distribution as dictated by the protocol. The easiness of protocol parameter modification in some wireless card has been previously addressed in [3] and [16]. To show the effect of non-cooperation, we simulated a simple 20 second scenario using OPNET. The load on all the nodes is the same. The packet inter-arrival rate of all nodes is exponential with mean of $0.01 sec$ and the packet size is exponentially distributed with mean $2048 bytes$. The wireless network consists of 8 nodes transmitting to the same destination. Direct Sequence Spread Spectrum is chosen at the physical layer with $CW_{min} = 32$ and $CW_{max} = 1024$ according to the standard [1]. The non-cooperating node in this case still chooses from a uniform distribution but with a fixed window size of 24. A non-cooperating node might still want to randomize to prevent being detected or avoid constant collisions with another non-cooperating node. We show the MAC delay experienced by one of the cooperating nodes and that of the non-cooperating in Fig. 2(a). In Fig 2(b), we also show the data dropped due to buffer overflow. Here we have considered a buffer of length $256 Kbits$. The non-cooperating node experienced an average data loss of $500 Kb/s$, whereas one of the cooperating nodes has a drop data rate of about $1.3 Mb/s$. This difference is a reflection of the difference in the node throughput at about $800 kb/s$, very significant considering the goodput of this scenario is less than $4 Mb/s$. Here we have only shown the results for one of the seven cooperating nodes as they all experience similar throughput and delay.

Several papers have addressed detection of protocol non-compliance, specifically with the backoff mechanism [16, 15, 5] and others have proposed some modifications to the backoff mechanism in order to make detection of noncooperation easier [5, 13]. DOMINO [16] first collects periodically backoff data during a monitoring period. After every monitoring period, it compares the backoff of a node to the nominal of the network with some tolerance parameter. DOMINO also keeps a cheating counter for every node that is incremented if a potential non compliance is detected and decremented if the data collected from a node passes the threshold test. If the counter reaches a threshold of K, the node in question is considered cheating. This detection scheme is not robust against more adaptive cheating mechanisms as mentioned by the authors. For example, by knowing the duration of the collection period, a non compliant node can follow the backoff mechanism of IEEE 802.11 for 3 periods so that its counter gets decremented at least twice and then follow a very short backoff during the next monitoring period, which may cause at most an increment of 2 in the cheating counter. Thus, the selfish node keeps the counter within bounds and avoids being detected. Another weakness of DOMINO is that no backoff measurements are collected after sensing a collision, thus allowing a selfish user to go undetected when transmitting with short backoff after a channel collision. It is hard to detect non-cooperation of nodes since the backoff times are of random nature, and a lot of statistics need to be detected before any assertion can be made. In general a selfish node can adapt its backoff time to the detection mechanism thus a detection mechanism will only limit the extent of non-cooperation.



(a) MAC Delay



(b) Packet Drop

Figure 2: Cooperation vs Non-Cooperation

3. BAYESIAN GAMES AND PROTOCOL DESIGN

In the game theory literature, what we have called a selfish user is considered to be merely a rational user, who wants to maximize his or her own utility, as one would expect. In our case for example, the utility of a user can be a function of the throughput and delay. Before we proceed further, we first introduce few definitions, concepts and results that we will need in the subsequent sections. When the payoffs of other players are not well known in advance or depend on the player types, the game is considered to have incomplete information. We thus resort to Bayesian games [7, 6]. An n player Bayesian game can be described as follows

$$\Gamma = \{S_1, \ldots, S_n, T_1, \ldots, T_n, p_1, \ldots, p_n, U_1, \ldots, U_n\}$$

where S_i is the set of strategies of player i. T_i is the set of types of player i. $p_i = p(t_{-i}|t_i)$ is player belief about other player types t_{-i} given his own type t_i. U_i is the player utility and is a function of the player types and their strategies.

An extension of Nash equilibrium in incomplete information games is Bayesian equilibrium. A strategy profile $\sigma = (\sigma_1, \ldots, \sigma_n)$ is a Bayesian equilibrium of Γ if

$$\sum_{t_{-i} \in T_{-i}} p(t_{-i}|t_i) U_i[\sigma(t), t] \geq \sum_{t_{-i} \in T_{-i}} p(t_{-i}|t_i) U_i[\sigma_{-i}(t), s_i, t], \forall i, s_i \in S_i \qquad (2)$$

where σ_i is the plan of action for each possible type.

$$\sigma_i : T_i \rightarrow S_i$$

In other words, and along the Nash equilibrium concept, no player wants to deviate from $\sigma_i(t_i)$ given his or her belief $p_i(t_{-i}|t_i)$ and that the other players are following the Bayesian equilibrium $\sigma_{-i}(t_{-i})$. We are ready now to revisit the random multiple access problem. For simplicity, assume that all users are of the same type, thus the Bayesian equilibrium (2) becomes

$$U_i[\sigma(t), t] \geq U_i[\sigma_{-i}(t), s_i, t], \forall i, s_i \in S_i \qquad (3)$$

3.1 Random Access Nash Equilibrium

We present the normal form game for three station games along the simple 2 station model presented in [17] and generalize the results to n station games. This will give insight into some of the findings in [14, 10, 9, 2] relying on different models. The station strategies are either Transmit or Wait, $S_i = \{T, W\}$. A successful transmission yields a payoff of u_s, a failed transmission due to collision yields a payoff of u_f and no transmission yields u_i. The payoffs are general

	T	W		T	W
T	u_f, u_f, u_f	u_f, u_i, u_f	T	u_f, u_f, u_i	u_s, u_i, u_i
W	u_i, u_f, u_f	u_i, u_i, u_s	W	u_i, u_s, u_i	u_i, u_i, u_i
	T			W	

Figure 3: 3 Stations' Normal Form Game

but must satisfy $u_f < u_i < u_s$ for obvious reasons. Let x, y and z denote the probability of transmission for station 1, 2 and 3 respectively. In order for user 1 to be willing to mix between transmitting and waiting, he must be indifferent to the payoff he gets from transmitting or from waiting. In other words $U_{1|T} = U_{1|W}$. $U_{i|X}$ is the expected utility of station i given it has followed strategy X.

$$U_{1|T} = U_{1|W} \Leftrightarrow \qquad (4)$$

$$yzu_f + (1-y)zu_f + y(1-z)u_f + (1-y)(1-z)u_s = u_i$$

We get symmetric equations when considering the other users. The solution of these sets of non-linear equations yield all the mixed Nash equilibria We are mainly interested in symmetric equilibrias due to fairness requirements and with $x = y = z$, (4) simplifies to

$$(u_s - u_f)x^2 + 2(u_f - u_s)x + u_s - u_i = 0 \qquad (5)$$

with unique solution

$$x^* = 1 - \sqrt{\frac{u_i - u_f}{u_s - u_f}}$$

In the general n station case, we get

$$(1-x)^{(n-1)} u_s + \sum_{k=1}^{n-1} \binom{n-1}{k} x^k (1-x)^{n-1-k} u_f = u_i$$
$$(1-x)^{(n-1)} u_s + (1 - (1-x)^{n-1}) u_f = u_i$$
$$\Rightarrow x_n^* = 1 - \left(\frac{u_i - u_f}{u_s - u_f}\right)^{\frac{1}{n-1}} \qquad (6)$$

Note that $u_i - u_f = c$ is the cost of transmission and $u_s - u_f = v$ is the payoff due to successful transmission. v can be associated to the valuation of the medium and/or packet. When transmission cost is negligible with respect to medium valuation, the probability of transmission is close to 1. This Nash Equilibrium will bring the network to a crawl, another instance of the tragedy of the common. On the other hand and as noted in [4], the backoff mechanism of IEEE 802.11 can be viewed as constant transmission probability in saturated state. This probability is a function of n, the number of stations, the contention window limits CW_{min} and CW_{max} and thus the protocol is not in equilibrium for a rational user to follow it. One way to regulate network performance is to add additional cost for transmission. However, the receiver cannot detect who transmits during a collision, thus we need to resort to a collision free scheme such as TDMA or FDMA to track and charge for transmissions. We will revisit the transmission costs and the success valuations in section 4.

3.2 The Revelation Principle

An important result relating to the Bayesian equilibrium that we will be using for resource allocation is the *revelation principle*:

Assume that $\sigma^*(t)$ is a Bayesian equilibrium of

$$\Gamma = \{S_1, \ldots, S_n, T_1, \ldots, T_n, p_1, \ldots, p_n, U_1, \ldots, U_n\}.$$

Then there exists a game

$$\Gamma' = \{S'_1, \ldots, S'_n, T_1, \ldots, T_n, p_1, \ldots, p_n, U'_1, \ldots, U'_n\}$$

such that in the new game Γ' truthful reporting of type is a Bayesian equilibrium. The strategy set $S'_i = T_i$ and the utility function is now $U'_i(s', t) = U_i(\sigma^*(s'), t)$ [7, 6, 12].

A mechanism with the strategy set equal the type set is called a *direct-revelation* mechanism. In summary, the revelation principle states that if the game Γ has an equilibrium strategy σ^*, then there exists a game Γ', as defined above, where reporting your type is the best strategy for every user given that others report their true type as well. The user type T_i in our problem corresponds to the user valuation of the time slot, the strategy set S_i could be a probability of medium access. The utility U_i is a function of nodes strategies, cost of transmission attempt and payoff. What the revelation principle allows us to do is instead of solving for the difficult Bayesian Nash equilibrium σ satisfying the set of equations (2), we can come up with an intuitive mechanism, by setting the proper utility function so as to make users report their true need for the medium.

A direct-revelation mechanism where truthful reporting is the best strategy is called *Incentive Compatible*. Thus, one of our objectives is to design a medium access protocol that is (i) *incentive compatible*. In developing an intuitive mechanism with a suitable utility function, we resort to auction theory as it has been extensively studied in the allocation of goods [12]. An important difference in our problem is that we are mainly after network performance and not seller (Access Point) utility maximization. The other requirement we have is (ii) *allocation efficiency*, that is assigning the time slots to those terminals valuing it the most. This constraint also provides quality of service in protocol design.

3.3 Truth Telling Second Price Auction

A clever and simple allocation mechanism where each player (bidder) wants to reveal his true valuation is the second-price auction. In the second-price auction, the seller has only one item for sale, and the highest bidder gets the item and only pays the second highest bid of the auction and not his own. Let v_i and b_i be player i value and bid for the item respectively. Bidder i utility is then

$$U_i(\mathbf{b}, v_i) = \begin{cases} v_i - max_{j \neq i} b_j & \text{if } b_i > \max_{j \neq i} b_j \\ 0 & \text{if } b_i \leq \max_{j \neq i} b_j \end{cases} \quad (7)$$

With this mechanism (utility), every bidder wants to bid his true value.

PROOF. Let x_i be user i bid and let $p_i = \max_{j \neq i} b_j$. User i wants to maximize his utility U_i. Let's now consider the case $x_i > v_i$, then we get

$$\begin{aligned} U_i =& P(p_i > x_i > v_i)0 + P(x_i > p_i > v_i)(v_i - p_i) + \\ & P(x_i > v_i > p_i)(v_i - p_i) \\ \leq & P(x_i^* = v_i > p_i)(v_i - p_i) \end{aligned}$$

By bidding $x_i^* = v_i$ we eliminate the second term which yields negative payoff without affecting the rest of the terms. A similar argument holds if user i were to bid $x_i < v_i$ $\square$

The winner payment is independent on his bidding price. The bidding price only determines the winner.

3.4 Vickrey Auction and Time Slot Allocation

The Vickrey auction adopts the idea of second price auction but applies when auctioning multiple items, say K. Each bidder submits his/her demand curve and the seller then calculates the aggregate demand on the goods to be allocated and the K highest winning bidders are assigned the goods. The winning bidders pay only the opportunity cost. The opportunity cost for user l refers to the value that other bidders would have paid if user l was not taking part in the auction. Formally, with K items to be allocated, each bidder $i \in \{1, \ldots, n\}$ submits a bidding vector $\mathbf{b_i} = (b_i^1, b_i^2, \ldots, b_i^K)$, where b_i^k is his valuation for a k^{th} item. Let $\mathbf{c_{-i}} = (c_{-i}^1, \ldots, c_{-i}^K)$ with element c_{-i}^l being the l largest value among $b_j^k, \forall k \in \{1, \ldots K\}, j \neq i$. The opportunity cost and the payment made by i for k_i items won can be expressed as

$$\sum_{m=1}^{k_i} c_{-i}^{K-k_i+m}.$$

This amount is the total value of the k_i highest losing bids, the opportunity cost. Vickrey auction is also incentive compatible, that is a node's best strategy is to bid its true valuation for the items. There are some practical problems with the Vickrey auction in certain settings and that's why it is not as widely used as sealed first price auction or ascending auction. However some variants of the Vickrey auction are very successful in practice. For example, Google AdWords uses it to auction advertisement slots next to search results[8].

Recall that our initial design criterion was to develop a medium access control protocol that runs in an environment where participating stations are individually owned and capable of altering protocol rules. Time slot allocation follows the idea presented in the Vickrey auction and time slots are assigned to the terminals that value them the most. Terminals participating in this protocol have an incentive to participate in the network and never deviate from reporting their true valuation for the medium. The base station must therefore collect the node valuation before assigning the time slots for transmission. Slot assignment is done in rounds. The number of time slots allocated in every round and the length of each time slot are design parameters and depend on the number of terminals associated to the AP, type of data traffic and supported services. This issue will be addressed in a later section. We can assume that at every round, K number of slots will be allocated to the active users, those who are associated with the receiver.

4. INCENTIVE COMPATIBLE MAC

The Incentive Compatible MAC (ICMAC) does not deal with the association and authentication mechanism, but we assume that a secure mechanism is in place. ICMAC is a TDMA based MAC and the receiver station has the task of scheduling the transmission of successfully associated stations. Fig. 4 summarizes the protocol operation. At the

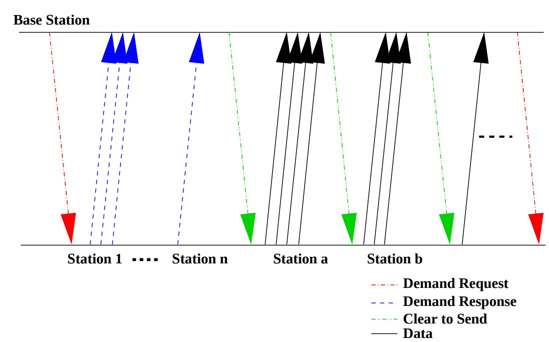


Figure 4: ICMAC Protocol

beginning of every round, the base station sends a Demand Request (DRQ) packet, to inform that it is taking bids for the K next time slots. Upon hearing a DRQ packet, every node responds with a Demand Response (DRS) packet. A DRS packet contains the station address, and its bids for each of the K time slots. Attributed to every station is an association ID (AID) and a demand response time slot. Thus during the bid collection time, the station access the medium in a deterministic TDMA fashion with no collision. After collecting all the demand curves, the base station aggregates the station demands to determine the winning K bids. Then sequential Clear To Send (CTS) messages are sent from the AP to the stations, from highest to lowest winning bids, informing them of the time of transmission and number of allocated successive transmissions. Along the CTS message, an optional acknowledgement is sent to the previous transmitting station on the previously sent data packets. In Fig. 4, station a is one of the n stations associated with the base station with the highest bids for that round. It receives a CTS packet informing it that it gets the next four time slots. After transmitting data for four successive time slots, station a listens for the next CTS packet to get an acknowledgment about its previously transmitted packets. A bit is associated with every previously transmitted packet for acknowledgment. In order to make the acknowledgment mechanism fruitful, the CTS message assigns no more than $MaxSch$ slots at a time. That is if a station wins more than $MaxSch$, the base station doesn't schedule all those transmissions in one shot, but breaks them apart, so they get progressively acknowledged.

4.1 Time Slot Valuation

A monetary or unit system has to be in place to carry out and enforce some of the ideas presented here. For the purpose of discussion, let *vDollar* be the network virtual currency. Thus every node i has a value $v_i(k)$ vDollar for a k^{th} time slot leading to the bidding vector $\mathbf{b_i}$. Terminals have a private value for the medium access, which is tightly dependent on delay and throughput. For example, the valuation of the time slot depends on packets present in the queue of the transmitter and/or running services such as VoIP. Packets are first categorized according to their type, for example data, voice, and video. These packet types have different bandwidth and delay requirements. The time slot valuation is a function of the waiting time and user/packet type. Three example profiles of packet valuation are presented herein and shown in Fig. 5. Every user is assumed to have independent valuation.

$$Y_l^1(t) = c_l$$

$$Y_l^2(t) = \begin{cases} a_l \exp(b_l t) + c_l, & t \in [0, t_l^{max}] \\ 0, & otherwise \end{cases}$$

$$Y_l^3(t) = c_l\left(\frac{1}{1+e^{-a_l(t-b_l)}}\right) + d_l$$

t represents the waiting time of the packet in the queue, l is the index of the packet type. a_l,b_l and c_l are type dependent parameters of the increasing valuation function. Note that t_l^{max} is also type defined. Some real-time applications might have hard constraints, and packets could be dropped if not transmitted before some expiration time t_l^{max}.

Another criterion that can also be considered is the ratio of packets in the queue with respect to the buffer size. When the queue size gets large, the new incoming packets might have to be dropped. In this case, the terminal node attributes an additional value to the time slot. Consider the following sigmoid valuation function that depends on the queue length L, the buffer size Q_{MAX}, and the packet position p.

$$W_l(p) = c_l\left(\frac{1}{1+e^{-a_l(p-b_l)}}\right) + d_l$$

The parameters c_l and b_l will be functions of $\frac{L}{Q_{MAX}}$. They are both increasing functions of $\frac{L}{Q_{MAX}}$. The parameter c_l determines the maximum increase in valuation of the time slot. b_l can be viewed as the limiting point of the affected packets. The longer the queue the more packets we want to send leading to increase in valuation. The function $W_l(p)$ decreases with the position of the packet in the queue. In

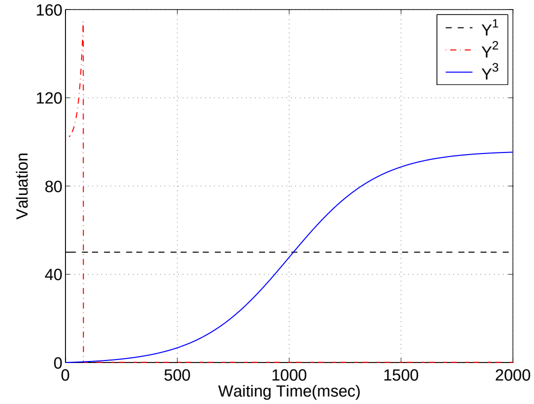


Figure 5: Valuation Function

Fig. 6, we show the additional valuation that is associated with the packet position for various queue lengths L for Q_{MAX}=100. Therefore the overall valuation of the time

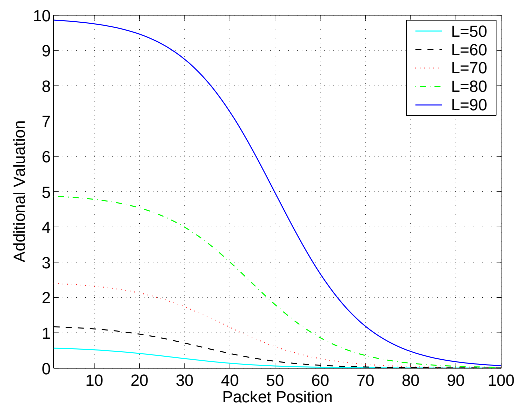


Figure 6: Queue Length Dependent Valuation

slot is a function of the packet waiting time, the packet position and the length of the queue. We are assuming that there are different queue types holding different packet types.

$$V_l(t,p) = Y_l(t) + W_l(p)$$

Note that the bidding/valuation vector can also be viewed as the inverse of the demand curve. Fig. 7 shows the demand curves of two terminals using the information present at their queues, or other information they might have about current running services. This information can also be simply represented in a vector. Quantization of the demand curve would also be used to shorten transmission of demand curves and simplify computation and decision making at the receiver. The receiver can calculate the aggregate demand and then allocate the time slot accordingly. In this case the number of time slots being offered is 20. As before the highest bids determine the winners and the price paid is the opportunity cost. The Vickrey auction requires that the bidding vectors be nonincreasing and this is usually satisfied for network users/stations.

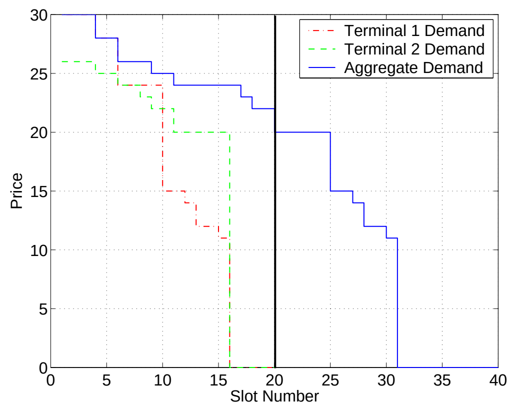


Figure 7: Demand Curves

4.2 Control Packets

ICMAC control messaging will be exchanged between the transmitters and the receiver to determine who will be transmitting and when. This control overhead must be analyzed thoroughly. The frame formats have been mainly borrowed from IEEE 802.11. The demand request (DRQ) packet and the clear to send (CTS) packet are all similar to the CTS of IEEE 802.11. The demand response (DRS) packet is similar to CTS as well with an additional field for the demand vector. The DRS has to be signed by the transmitter as well. The number of slots per round and the fragment size can be either advertised during association or through the DRQ packet. As mentioned above, we have not addressed neither the association mechanism nor the authentication mechanism, but they are both required for the ICMAC protocol. It is very important for these mechanisms to be safe as users will be paying for the service they receive.

5. ICMAC PERFORMANCE AND DESIGN PARAMETERS

Before we proceed further we define some parameters and tabulate packet sizes and design parameters in table 1. With little abuse of notation $phyhdr$ is shown in μs and in $bits$ and kept the same for $1Mb/s$ and $11Mb/s$ transmission rates. Design variables need to be chosen by an administrator based on the type of traffic that will be using the AP. The parameters designated will impact the overall throughput, delay and overhead. The control packets, DRQ, DRS, CTS are all sent at control transmission rate of $1Mb/s$ and the data packet is sent at either $1Mb/s$ or $11Mb/s$. We calculate the throughput of the protocol for what we consider reasonable parameters for some applications. We assume data occupy the whole fragment in this initial calculation. We will revisit performance after we address the design parameters.

$$Throughput = \frac{K * DATA}{RoundDuration}$$

In calculating the round duration we have to consider the transmission rate of the control and data packets.

Parameter	Value	Unit
Inter frame duration	$SIFS = 10$	μs
Physical layer delay	$phyhdr = 192$	μs
MAC header	$machdr = 272$	bits
Slots per round	K (design parameter)	slots
Fragment length	$FLength$ (design parameter)	bits
Value representation	$BidRep$	bits
DRQ packet length	$phyhdr + 160$	bits
DRS packet length	$phyhdr + 160 + KBidRep$	bits
Max packets scheduled	$MaxSch$	n/a
CTS packet length	$phyhdr + 160 + MaxSch$	bits
DATA packet length	$272 + FLength$	bits

Table 1: Frame sizes and Parameters

Parameter	Value
n	20
K	50
$frag\_size$	8192
$BidRep$	8

Table 2: Example Parameters

$$\begin{aligned} &RoundDuration \\ &= \frac{DRQ}{CtrlRate} + SIFS + n(\frac{DRS}{ctrlRate} + SIFS) \\ &+ K(\frac{CTS}{CtrlRate} + SIFS + \frac{DATA}{DataRate} + SIFS) \end{aligned}$$

A new incoming packet of highest type arriving after the DRQ transmission has to wait for the remaining time of the round duration plus the new bid collection time. Recall that stations submit bids only when they have traffic to send or some services running, such as VoIP. The round duration is $82ms$ for $11Mb/s$ data rate with control rate kept at $1Mb/s$. The throughput is $878Kbits/s$ and $4.981Mb/s$ for the parameter set given in Table 2 with a data rate of $1Mb/s$ and $11Mb/s$ respectively.

The performance drops with the number of stations due to bid collection at every round. In order to reduce the overhead incurred from this bid collection in large wireless network, the network designer can increase the number of slots allocated at every round. The other alternative is to auction multiple rounds at a time. The later option is also appropriate in situations where the services running in the network require sustainable throughput over multiple rounds. In Fig. 8 we show the potential throughput gain from auctioning multiple rounds at a time. One drawback to auctioning many rounds is that the maximum waiting time for highest type station will increase even when in general the average waiting time will decrease. Another drawback is that some slots may be wasted as the winning stations may have no packets to transmit at later rounds. The extreme case of allocating slots over multiple rounds becomes a fixed TDMA scheme which is not appropriate in data networks. We also plotted the round duration.

As ICMAC is a TDMA based access control and the slot sizes are fixed, the network designer has to choose properly the slot length and the number of slots auctioned at each round. We consider a time slot to contain a CTS control message, all the interframe durations and the data packet.

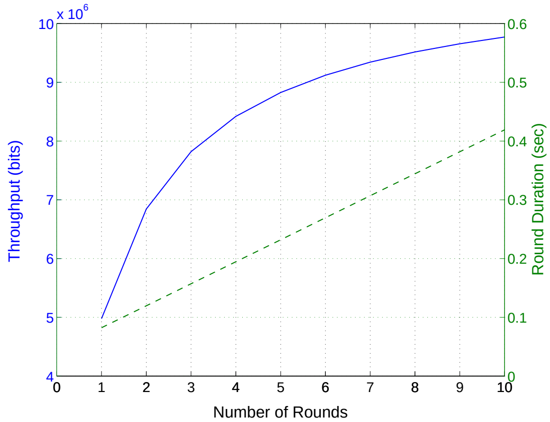


Figure 8: Throughput and Round Duration

Refer to Fig. 9 for better understanding. The overhead of a time slot is

$$h = 2SIFS + \frac{CTS}{CtrlRate} + phyhdr + \frac{machdr}{DataRate}$$

$h = 788\mu s$ and $h = 584.4\mu s$ for a transmission data rate of $1Mb/s$ and $11Mb/s$ respectively. We also denote by H, the round overhead associated with bid collection. It can be expressed as

$$H = SIFS + \frac{DRQ}{ctrlRate} + n(SIFS + \frac{DRS}{CtrlRate})$$

Recall that DRS size depends on K, the number of slots allocated per round, and $BidRep$, the number of bits representing a bid. For $n = 10$, $K = 50$ and $BidRep = 8$, $H = 7.982ms$

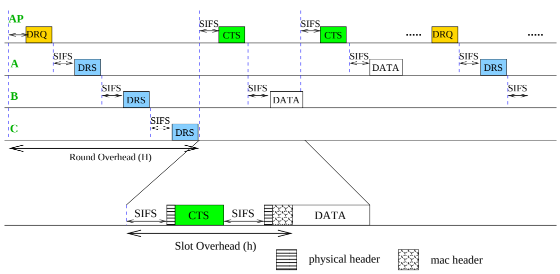


Figure 9: ICMAC Overhead

5.1 Fragment Size

As messages might be sent over multiple slots and data might not occupy the full time slot, we need to optimize transmission efficiency with respect to time slot duration. Clearly the optimum slot duration will be a function of the message length and the overheads h and H. We assume that the data size is distributed according to $f(x)$. The problem of using the fixed slot size efficiently becomes:

$$\min_Y \int_0^\infty (Y + \frac{H}{K}) \lceil \frac{x}{Y-h} \rceil f(x)\, dx \qquad (8)$$

$$\Leftrightarrow \min_Y (Y + \frac{H}{K}) \int_0^\infty \lceil \frac{x}{Y-h} \rceil f(x)\, dx.$$

In (8), Y is the slot duration, $\lceil \frac{x}{Y-h} \rceil$ is the number of slots required by a message of length x, $\frac{H}{K}$ represents the per transmission overhead due to the round overhead H. With $Z = Y + \frac{H}{K}$ and $h' = \frac{H}{K} + h$, the optimization (8) can be rewritten as

$$\min_Z Z \int_0^\infty \lceil \frac{x}{Z-h'} \rceil f(x)\, dx. \qquad (9)$$

Now consider only the integral term of equation (9):

$$\begin{aligned} &\int_0^\infty \lceil \frac{x}{Z-h'} \rceil f(x) dx \\ &= \sum_{k=1}^{\infty} \int_{(k-1)(Z-h')}^{k(Z-h')} k f(x) dx \\ &= \sum_{k=1}^{\infty} k \mathbb{P}((k-1)(Z-h') < X \leq k(Z-h')) \qquad (10) \end{aligned}$$

$\mathbb{P}((k-1)(Z-h') < X \leq k(Z-h'))$ is the probability that a message m requires k time slots. As an example, we first look at exponentially distributed packet lengths, and then exponentially distributed mixed with constant packet lengths.

5.1.1 *Exponential distribution*

With message length exponentially distributed with mean $\bar{m}$, (10) can be expressed as

$$\begin{aligned} &\sum_{k=1}^{\infty} k(exp(-\frac{(k-1)(Z-h')}{\bar{m}}) - exp(-\frac{k(Z-h')}{\bar{m}})) \\ &= \sum_{k=0}^{\infty} exp(-\frac{k(Z-h')}{\bar{m}}) = \frac{1}{1 - exp(-\frac{(Z-h')}{\bar{m}})} \qquad (11) \end{aligned}$$

The minimization (9) becomes

$$\min_Z \frac{Z}{1 - exp(-\frac{(Z-h')}{\bar{m}})} \qquad (12)$$

with the solution satisfying

$$exp(\frac{Z-h'}{\bar{m}}) - (1 + \frac{Z}{\bar{m}}) = 0. \qquad (13)$$

(13) has a unique solution $Z > h'$ that can be easily found numerically. The solution Z^* corresponds to a time duration which can be translated to data fragment size of

$$frag^* = (Z^* - h') * DataRate.$$

In the case where all packets belonging to the same message are scheduled with one CTS because they have the same value, the transmission efficiency problem stays the same, but now

$$h = SIFS + phyhdr + \frac{machdr}{DataRate}.$$

$Maxsch$ is disabled here. We plot in Fig. 10 the fragment size solution with respect to mean packet size $\bar{m}$ for n=10, K=50, fixed control transmission rate of $1Mb/s$ and data transmission rates of $1Mb/s$ and $11Mb/s$. We have included results on both individual packet scheduling and multiple packet scheduling. The solution for optimum packet size is smaller for multiple scheduling than individual scheduling since the fragmentation penalty is less significant.

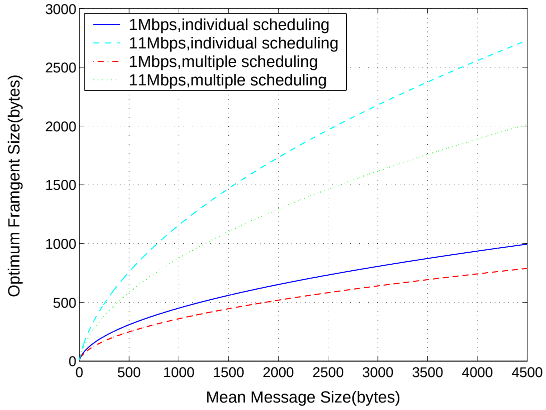


Figure 10: Optimal Data Fragment Size

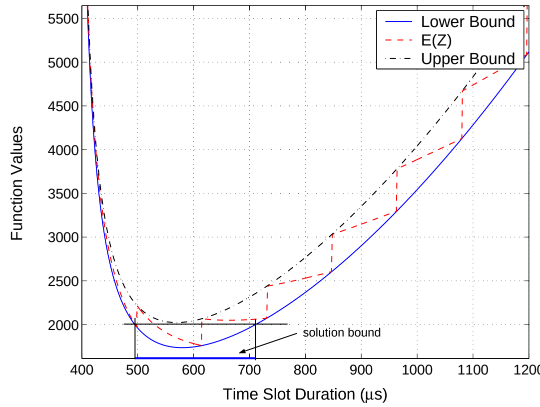


Figure 11: Mixed Traffic Optimum Solution

5.1.2 Mixed exponential and constant size messages

We assume that traffic with exponentially distributed message size is sent with probability p and traffic with constant message size is sent with probability $1-p$. $\bar{m}$ is the mean of the exponential distribution and $\bar{v}$ is the constant message size. (9) becomes

$$\min_Z Z(p\frac{1}{1-exp(-\frac{(Z-h')}{\bar{m}})} + (1-p)\lceil\frac{Z-h'}{\bar{v}}\rceil) \qquad (14)$$

The above problem is not convex; however, we can find a solution bound using (15).

$$\begin{aligned} &Z(p\frac{1}{1-exp(-\frac{(Z-h')}{\bar{m}})} + (1-p)\frac{Z-h'}{\bar{v}}) \leq \\ &Z(p\frac{1}{1-exp(-\frac{(Z-h')}{\bar{m}})} + (1-p)\lceil\frac{Z-h'}{\bar{v}}\rceil) \leq \qquad (15) \\ &Z(p\frac{1}{1-exp(-\frac{(Z-h')}{\bar{m}})} + (1-p)(\frac{Z-h'}{\bar{v}}+1)) \end{aligned}$$

The minimum of the upper bound function in (15), is an upper bound on the minimum of the solution. Now using the lower bound in (15) we can limit the range of the solution. We depict all three functions to show the process with which we find a solution bound in Fig. 11. The figure shows the case of p=0.5, $\bar{v} = (160*8/11E6)s$ and $\bar{m} = (1024*8/11E6)s$. The solution in this case is $614\mu s$ for the slot duration translating to $319bytes$ for the data fragment size. For p=0.75, we get $480bytes$ for the data fragment size, as more messages are distributed according to the exponential distribution.

In addition to the message distribution, another important constraint that the designer needs to keep in mind is that of the physical medium. The longer the fragment, the more susceptible it is to errors. Thus there are different limits to the fragment length in different environments.

5.2 Number of Slots per Round

Recall that we expressed round duration as

$$RoundDuration = H + K*(h+\frac{frag}{DataRate}) = KZ$$

The round duration is tied with connection setup time. The round duration time must be appropriate for the traffic supported on the network. For example, for delay sensitive application with constant bit rate, stations rather have the slots spread through the round instead of getting all the transmissions in one shot. In the current protocol, the AP schedules only according to the aggregate demand, thus limiting the round duration would allow interleaving between station transmissions.

6. SIMULATION RESULTS

We now show some simulation results to confirm analytical solutions obtained in previous sections and we also present comparative performance figures between ICMAC and the IEEE 802.11 DCF for few simple scenarios. The focus here has been the network throughput and nodes are always in saturation mode. We have used OPNET for simulation and each point corresponds to multiple runs. The scenarios are all the same and that is n nodes sending to one AP.

6.1 ICMAC Design

We have addressed design issues in section 5 relating to optimal fragment size and number of slots per round. We now show results for n=10 and K=50 nodes with $11Mb/s$ data rate and exponentially distributed message for 4 different means of 512 $bytes$, 1024 $bytes$, 2048 $bytes$ and 4096 $bytes$. Packets are individually scheduled, $Maxsch$=1. We plot in Fig. 12 the throughput for different fragment sizes and the network performance peeks are in agreement with the analytical optimum fragment size of 771 $bytes$, 1174 $bytes$, 1756 $bytes$ and 2591 $bytes$ for message mean of 512 $bytes$, 1024 $bytes$, 2048 $bytes$ and 4096 $bytes$ respectively. Simulation results for optimal fragment size in the case of multiple packet scheduling is also in agreement with the analytical solution.

6.2 Multiple Packet Scheduling

As previously mentioned there is an advantage for scheduling multiple transmissions for the same station with only one CTS packet. This benefit is highlighted in Fig. 13 again for n=10 and K=50 for different message means. The throughput gain experienced in this scenario is between four to five percent. The main overhead comes from bid collection as it depends on the number of nodes and the value representation of the K values. One way to reduce this overhead is to

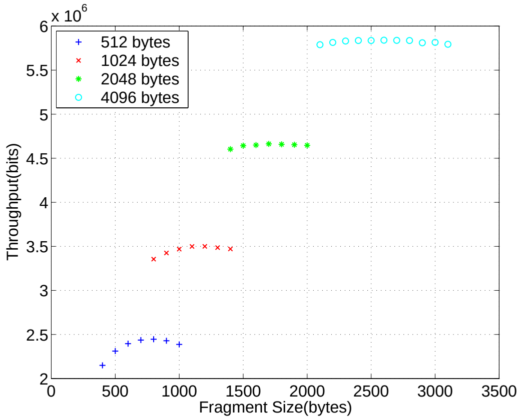


Figure 12: Throughput vs Fragment Size

collect station bids for multiple rounds. The winning station gets the same slot over multiple rounds. We can still use the optimum fragment solution as before but now $\frac{H}{K}$ is divided by the number of rounds auctioned at every stage. In Fig. 14, we show the throughput gain of collecting bids every 2 rounds versus collecting bids at every round. We have used the same simulation scenario with exponential message size.

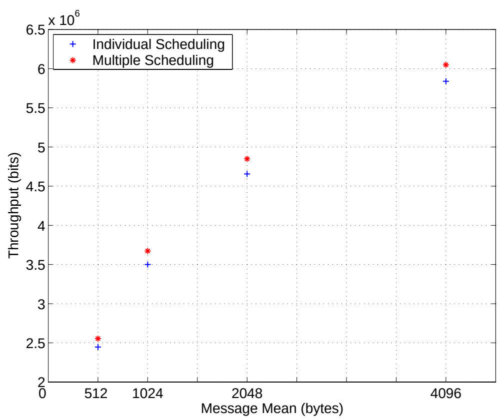


Figure 13: Multiple Scheduling

6.3 ICMAC vs IEEE 802.11

We finally show a throughput comparison in Fig. 15 between ICMAC and IEEE 802.11 using simulation results. We have simulated 5 runs of 30 seconds for each point. For IEEE 802.11, we used a simple collision model, that is a transmission is lost only if two or more transmissions collide. The RTS/CTS threshold is set to $512bytes$, that is packets larger than $512bytes$ exchange control messages before data transmission. Packets larger than $2304bytes$ are fragmented. Data and control transmission rates are at $11Mb/s$ and $1Mb/s$ respectively for both protocols. All nodes are in saturation mode and message sizes are again exponentially distributed with varying means as indicated in the figure. Note that we are not favoring ICMAC here with exponential message size since the fragments are of fixed size and

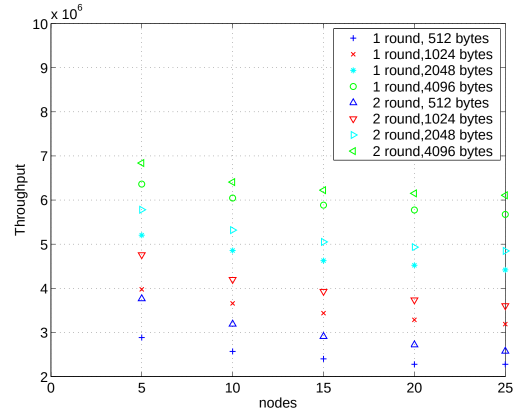


Figure 14: Multiple Round Scheduling

some bandwidth is wasted when data does not fully occupy the slot. We show the results for ICMAC with 2 round scheduling and K=50. ICMAC performs better with larger message means and as expected the throughput drops with the number of nodes due to the initial bid collection at the beginning of every round. IEEE 802.11 shows similar trend with the number of nodes but at a slower rate.

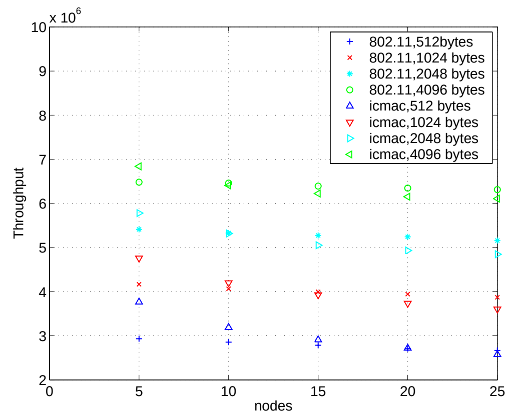


Figure 15: IEEE 802.11 and ICMAC Throughput

7. CONCLUSION

In this work, we have introduced a new Incentive Compatible Medium Access Control that takes into account the independence of the entities participating in the network. The stations share a common bandwidth resource and have a utility maximization objective. Using game theory and the Nash equilibrium concept, we showed why the current IEEE 802.11 standard is not appropriate for such a scenario as it is based on random access. We have showed in a homogenous scenario, where the nodes have the same valuation, that the symmetric Nash equilibrium yields a very low network throughput due to the low cost to payoff ratio of transmissions. The initial objective of this work has been to design a new MAC protocol that is incentive compatible, where participating stations have an incentive to follow protocol rules and the Nash equilibrium is to report true type and access valuation. We have resorted to auction theory to allocate bandwidth in a non-cooperative environment. The new Incentive Compatible MAC is based on the Vickrey auction. First bids are collected from the various stations, and then transmission time slots are assigned to the various stations according to the highest bids. The price paid by the winning station reflects the opportunity cost. The benefit of using Vickrey auction is two fold. First, it keeps the bidding strategy simple even in a incomplete information setting, as nodes only know their type and not that of the other competing terminals. The other important feature of Vickrey auction relates to setting up the appropriate transmission cost. The transmission cost is self adjusting and set by the competing users according to the network load and demand curves. The low transmission cost was the network deteriorating factor in random access MAC. No administrator is required to adjust the usage price according to the load. The wireless network usage becomes free under light traffic load and those who do not wish to pay for bandwidth can still use it then. In addition to being robust to greedy behavior, ICMAC shows no degradation in performance with respect to IEEE 802.11 for realistic network size. ICMAC shows great potential as we have not fully explored other potential improvements. We have tried to keep many parameters similar to IEEE 802.11 for comparative reasons. For instance note that after bid collection, there is no need to individually send a CTS, the AP can broadcast all at once the slot allocation to all the associated nodes. In this scenario, acknowledgment would be left to higher layers as CTS no longer transmits acknowledgment bits for the previously transmitted packets.

8. REFERENCES

[1] *IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, 1999.

[2] E. Altman, R. E. Azouzi, and T. Jiménez, “Slotted Aloha as a stochastic game with partial information,” in *Proc. Wiopt*, 2003.

[3] J. Bellardo and S. Savage, “802.11 denial-of-service attacks: Real vulnerabilities and practical solutions,” in *Proc. USENIX Security Symposium*, August 2003.

[4] G. Bianchi, “Peformance analysis of the IEEE 802.11 distributed coordination function,” *IEEE journal on Selected Areas in Communications*, vol. 18, no. 3, pp. 535–547, March 2000.

[5] A. Cardenas, S. Radosavac, and J. Baras, “Detection and prevention of MAC layer misbehavior for ad hoc networks,” in *Proc. Workshop on Security of AD Hoc and Sensor Networks (SASN)*, 2004, pp. 17–22.

[6] P. Cramton, *Advanced Microeconomics Lecture Notes*, 2005.

[7] D. Fudenberg and J. Tirole, *Game Theory.* MIT Press, 1991.

[8] Adwords learning center. Google. [Online]. Available: http://www.google.com/adwords/learningcenter/

[9] Y. Jin and G. Kesidis, “Equilibria of a noncooperative game for heterogeneous users of an ALOHA network,” *IEEE Comm. Letters*.

[10] ——, “A pricing strategy for an ALOHA network of heterogenous users with inelastic bandwidth requirements,” in *CISS*, March 2002.

[11] J. Konorski, “Playing CSMA/CA game to deter backoff attacks in ad hoc wireless LANs,” in *Proc. ADHOC-NOW*, 2005, pp. 127–140.

[12] V. Krishna, *Auction Theory.* Academic Press, 2002.

[13] P. Kyasanur and N. Vaidya, “Detection and handling of MAC layer misbehavior in wireless networks,” in *Proc. International Conference on Dependable Systems and Networks*, 2003, pp. 173–182.

[14] A. B. Mackenzie and S. B. Wicker, “Stability of multipacket slotted Aloha with selfish users and perfect information,” in *Proc. IEEE Infocom*, vol. 3, 2003, pp. 1583–1590.

[15] S. Radosavac, J. Baras, and I. Koutsopopoulos, “A framework for MAC protocol misbehavior detection in wireless networks,” in *Proc. ACM Workshop on Wireless Security (Wise05)*, 2005, pp. 33–42.

[16] M. Raya, J.-P. Hubaux, and I. Aad, “DOMINO: A system to detect greedy behavior in IEEE 802.11 hotspots,” in *Proc. MobiSys*, 2004, pp. 84–97.

[17] Y. Xiao, X. Shan, and Y. Ren, “Game theory models for IEEE 802.11 DCF in wireless ad hoc networks,” *IEEE Radio Communications*, Mar. 2005.