UC Santa Cruz

UC Santa Cruz Previously Published Works

Title

VCMA: Efficient Channel Access in Wireless Mesh Networks Using Virtual Coordinates

Permalink

<https://escholarship.org/uc/item/6q5251bd>

Author

Garcia-Luna-Aceves, J.J.

Publication Date

2012-10-29

Peer reviewed

VCMA: Efficient Channel Access in Wireless Mesh Networks Using Virtual Coordinates

Ashok N Masilamani, J.J. Garcia-Luna-Aceves
Department of Computer Engineering
University of California, Santa Cruz
Santa Cruz, CA 95064
Email: jj, ashok@soe.ucsc.edu

Abstract—The Virtual Coordinate Multiple Access (VCMA) protocol is introduced and analyzed. VCMA is a medium access control protocol that defines transmissions schedules dynamically based on the assignment and exchange of virtual coordinates relative to an elected root node in a mesh network. VCMA is shown to attain feasible transmission schedules within a finite time, and its channel access delay properties are analyzed. The performance of the VCMA is compared with the performance of 802.11 DCF, which is a contention based MAC protocol; the five-phase reservation (FPRP) protocol, which is a schedule-based MAC protocol based on reserving time slots over fixed-length frames; and the node activation multiple access (NAMA) protocol, which is representative of distributed transmission scheduling based on probabilistic elections per slot. The performance comparison shows that VCMA attains much higher throughput than 802.11 DCF, FPRP or NAMA, and that it has lower variance in channel access intervals than contention-based schemes and schedule-based protocols based on probabilistic elections or reservations carried out in fixed-length frames.

I. INTRODUCTION

The price, performance and form factors of sensors, processors, storage elements, and radios today are at a point that autonomic wireless networks embedded in the environment and connecting mobile people with other people, information, services, sensors, and local or remote devices are finally possible. However, realizing this is far from being feasible in practice. We argue that, at least in part, this is due to the inherent limitations in the basic designs of medium access control (MAC) protocols that have been used in wireless mesh networks over the past 40 years, and which are derivatives of protocols originally designed for wired networks.

Contention-based MAC protocols, including IEEE 802.11 distributed coordination function (DCF) [1], force senders to back off on a packet-by- packet basis in the presence of multiple access interference (MAI), which has very negative consequences on throughput and channel access delays. On the other hand, as our summary of related work presented in Section II indicates, MAC protocols based on the establishment of transmission schedules typically organize the channel into transmission frames consisting of a fixed number of time slots assigned to nodes through elections or reservations. While these protocols eliminate the problems associated with back-offs after noise of interference is detected, their use of probabilistic methods for electing and reserving time slots in a fixed-length frame lead to the *coupon collector's*

problem [2], which relates to the long time taken for each node in a neighborhood to obtain a feasible time slot for its transmissions (i.e., “all coupons are collected”), as well as the variance of the time between two consecutive transmissions by the same node. This time taken and the variance is proportional to the two-hop neighborhood size of a node. For example, in a two-hop neighborhood of 30 nodes, 120 time slots are needed on average for all the nodes to access the channel (the expected time to collect n coupons is $T(n) = nH_n$ where H_n is the Harmonic number).

Section II summarizes prior related work. As our summary indicates, whether MAC protocols are based on contention or the establishment of transmission schedules, all these protocols use node identifiers as the basis for assigning the channel. We propose a new approach to collision-free channel access in this paper, which we call VCMA (Virtual Coordinate Multiple Access) and which uses virtual coordinates for scheduling, rather than node identifiers.

Section III describes VCMA, which consists of a distributed algorithm used to establish virtual coordinates, a transmission scheduling algorithm that operates on these virtual coordinates, and a neighbor discovery algorithm with which nodes exchange the virtual coordinates of their immediate neighbors. The transmission schedules established by VCMA eliminate the problems associated with back-offs common in contention-based MAC protocols and at the same time, the use of virtual coordinates reduces the coupon collector's problem and the large variances in channel access intervals that are common in probabilistic election schemes like NAMA [3]. VCMA takes advantage of the simple observation that a collision occurs only when two nodes within two hops transmit in the same time slot and that hop-ordering combined with time-slot numbers can be used to avoid such collisions. VCMA establishes virtual coordinates by means of hop-ordering of nodes from a root node elected through simple hop-by-hop signaling. While virtual coordinates based on hop-distance metrics have been used extensively for routing in the past, to the best of our knowledge, VCMA is the first approach that uses virtual coordinates for distributed transmission scheduling. In this paper, we focus on wireless mesh networks with static topologies which can take full advantage of using virtual coordinates rather than node-IDs for channel access.

Section IV analyzes the performance of VCMA in terms

of goodput and channel access delays. The results show that VCMA is a better approach in terms of all performance metrics compared to contention-based channel access represented by 802.11, as well as schedule-based channel access based on either elections (as in NAMA [12]) or reservations (as in FPRP [9]) of time slots selected from a fixed-length transmission frame, which has been the norm in prior schedule-based MAC protocols.

II. RELATED WORK

Many medium access control (MAC) protocols have been proposed to control access to a common wireless channel using contention and distributed scheduling schemes. Because of space limitations and the performance limitations of contention-based schemes in the presence of hidden terminals, we focus our survey of prior work on schemes aimed at distributed transmission scheduling. What is striking about all of this prior work is that these schemes organize the communication channel into fixed transmission frames consisting of a pre-defined number of time slots used to access the channel.

Dynamic transmission scheduling schemes can be categorized into topology-independent and topology-dependent schemes. Topology independent schemes (e.g., [5]) establish transmission schedules dynamically and such schedules are independent of the topology of the network. The times when node i transmits in a frame corresponds to a unique code such that, for any given neighbor k of i , node i has at least one transmission slot during which k and none of the neighbors of node k are transmitting. This topology independence makes the transmission schemes very robust in the presence of node mobility; however, Kunz and Rentel [6] have shown that this approach has similar performance to that of slotted ALOHA protocols.

Topology-dependent transmission scheduling protocols attempt to establish transmission schedules taking into account the connectivity of the network and in some cases the traffic at each node. The assignment of time slots to nodes is based either on the election of entities competing for the data time slots (nodes or links), or the selection of reservation requests for data time slots according to a set of predefined rules. Some schemes require an initial topology-independent schedule, followed by some negotiation among network nodes used to obtain a final schedule [[7], [8]]. In topology-dependent scheduling protocols based on reservations, the channel is divided into frames consisting of a fixed number of time slots, and each time slot is divided into several mini-slots dedicated for the contention and reservation of the time slots as well as the transmission of data in the time slot. FPRP [9] and HRMA [10] are examples of such protocols that have frames with fixed number of time-slots.

There are many examples of topology-dependent transmission scheduling protocols based on the election of transmission schedules in a distributed manner. To elect transmission schedules, each node knows the identities of all other nodes one and two hops away from itself, and the present time in the network [11], [12], [4]. Depending on the protocol, nodes use

a contention-based approach like slotted ALOHA during the control section of a frame to communicate to their neighbors either the identifiers of their own neighbors and themselves, or the identifiers of the links to their own neighbors. Each node builds and maintains a list of contending entities (nodes or links) and uses this list to determine which node should be given access to the channel during each time slot of the data section of the frame. To accomplish this task, the node applies a permutation function on the list of contending entities to select a winning node from the list of nodes for each time slot of the transmission frame. The main limitation of all the above protocols is that when the number of nodes in the two-hop neighborhood increase, the time taken for all nodes to access the channel at least once becomes very high.

Considerable work has been reported on the establishment of efficient transmission schedules in a distributed manner taking into account the nodal traffic demands and attempting to limit the overhead incurred in the establishment of schedules that approach the optimum [13], [14]. However, none of the approaches reported to date are practical in wireless networks because of the extensive signaling they incur.

There have been a number of location-based MAC protocols proposed in the past (eg., [18], [19]), but all these MAC protocols use GPS location to assist media access. Furthermore, there has been extensive research in the past on location based routing (e.g., [17]). However, to the best of our knowledge, no prior proposals exist for the establishment of transmission schedules based on virtual coordinates and VCMA is the first such protocol.

III. VCMA

VCMA achieves contention resolution among nodes for broadcast transmissions over a single common channel by establishing virtual coordinates for the nodes. To accomplish this, VCMA assumes that: the radios used in the network are half-duplex and can tune to only one channel at a time; each node in the network is assigned a node identifier (nid); the radio links used for communication in the network are bidirectional; channel access time is slotted and time slots have a fixed duration; and any pair of nodes can be synchronized at the time-slot level. These assumptions are similar to those used in MAC protocols based on transmission scheduling. The time slotting needed in VCMA can be attained in practice using such distributed clock synchronization schemes as those demonstrated in the past by Rentel and Kunz [15] and Djukic and Mohapatra [16]. Though tight synchronization may be difficult and expensive to achieve in some networks, the protocol does not have any bounds on the slot duration used in the network. Hence the slot duration can be adjusted to suit network synchronization capabilities without affecting the throughput of the network. As such, the rest of this paper simply assumes that time slotting is available.

For simplicity of exposition, the rest of this paper assumes that multiple access interference (MAI) occurs only among one and two-hop neighbors, which is also assumed in most MAC protocols based on transmission scheduling. Accordingly, the

neighborhood of a node is assumed to consist of those nodes whose transmissions the node can decode, which we call one-hop neighbors, and the one-hop neighbors of those nodes, or two-hop neighbors. In practice, MAI may occur due to nodes more than two hops away. This can be taken into account in VCMA very easily by using modulo-4 operation rather than modulo-3 in the transmission algorithm that is explained in Section III-B. The modulo operation is denoted by the % sign in the rest of the paper and $a\%b$ is the remainder of division of number a by b .

Each time slot in VCMA is used for the transmission of a *Hello* packet and zero or more data packets. The *Hello Packet* contains an entry for each one-hop neighbor of the transmitting node and the node itself. Each entry consists of the node identifier of the node and its virtual coordinates. Contention resolution at each time-slot is achieved based only on the two-hop neighborhood information available at every node.

A. Virtual Coordinates

The virtual coordinates for a node consist of a tuple of four fields, as shown in Figure 1. *Root-A* and *Root-B* are the root nodes of the Directed Acyclic Graph that *Label-A* and *Label-B* belong to respectively. *Label-A* is an integer equal to the hop-distance from the root node of the entire connected part of the network that the node belongs to, which is the node with the smallest node-ID. *Label-B* is the hop distance from the node with the smallest node-ID that has the same *Root-A*, *Label-A* and *Root-B* values as the node.

When a node is switched on, it names its own node-ID as *Root-A* and *Root-B* and assigns zero to *Label-A* and *Label-B*. It starts updating the coordinates on receiving *Hello Packets* from its one hop neighbors. Algorithm 1 is the procedure used to update the coordinates. It is executed every time a new *Hello* packet is received from a neighboring node. Figure 3-(a) shows the virtual coordinates for a network that is arranged in a grid formation. For this network, *Root-A* is 1 for all the nodes. *Root-B* values are 2, 6, 10, 14 and 18 for the nodes in rows 2, 3, 4, 5 and 6 respectively. The virtual coordinates for a randomly distributed network is shown in Figure 3-(b)

Fig. 1. Virtual coordinates(VC)

B. Transmission Algorithm

Fig. 2. Frames = 9 slots

A time slot is the smallest time unit for transmitting one or more complete data packets. It is assumed that the time

Algorithm 1 Assignment of Virtual Coordinates

```

1: for all Neighbors nbr in one.hop.neighborhood, do
2:   if nbr.RootA == node.RootA then
3:     if nbr.LabelA < node.LabelA - 1 then
4:       node.LabelA ← nbr.LabelA + 1
5:       Flag = 1 //Indicates change in LabelA
6:     end if
7:   else if nbr.RootA < node.RootA then
8:     node.LabelA ← nbr.LabelA + 1
9:     node.RootA ← nbr.RootA
10:    Flag = 1
11:   end if
12:   if Flag == 1 then
13:     node.LabelB ← 0
14:     node.RootB ← node.ID
15:   end if
16: for all Neighbors nbr in one.hop.neighborhood, do
17:   if nbr.RootA == node.RootA && nbr.LabelA ==
    node.LabelA then
18:     if nbr.RootB == node.RootB then
19:       if nbr.LabelB < node.LabelB - 1 then
20:         node.LabelB ← nbr.LabelB + 1
21:         node.RootB ← nbr.RootB
22:       end if
23:     else if nbr.RootB < node.RootB then
24:       node.LabelB ← nbr.LabelB + 1
25:       node.RootB ← nbr.RootB
26:     end if
27:   end if
28: end for
29: end for

```

slots are synchronized and that the slots across all nodes in the two-hop neighborhood have the same time-slot number at any particular instant. Nine consecutive time slots form a *frame* and each of these time slots form a *pool*. The notion of *frame* used in VCMA is different from the ones used in other scheduling protocols, where the number of slots in a frame are predefined for the network. The transmission algorithm executed at every slot in all nodes is given in Algorithm 2. $(Slot.number\%9) + 1$ is calculated at the start of each slot in order to determine the *pool* to which the particular time slot belongs. The nodes that can contend for a particular time slot are determined by the labels of the nodes. From its label, a node calculates $LabelA\%3$ and $LabelB\%3$ and from the values obtained, determines the *pool* to which it belongs. Figure 3-(c) shows the *pools* and the corresponding $LabelA\%3$ and $LabelB\%3$ values for the nodes in the network in Figure 3-(a).

If the node can contend for the slot, then it enters a hash-based election with other nodes in the two-hop neighborhood that have the same $LabelA\%3$ and $LabelB\%3$ values, called contending nodes. For the network shown in Figure 3-(a), none of the nodes have a contending node in their slot. But in Fig. 3-(b), it can be seen that nodes 16 and 17 have the same labels and therefore, contend for the same slot in the frame for that neighborhood. If there are no other nodes in the two-hop neighborhood that have the same $LabelA\%3$ value as the current node, then the node transmits in all three slots in the frame that belong to the particular $LabelA\%3$ value. For example, in Fig. 3-(b), node 6 is the only node in the two hop neighborhood that has $LabelA\%3$ value as 0. So node 6 transmits in slots belonging to pools 4 and 7 in addition to its original pool 1. Also, an exception is made in the algorithm for the node with both $LabelA\%3$ and $LabelB\%3$ values of 0

Fig. 3. Virtual Coordinates and Pools

zero (Node 1 in Figs. 3-(a) and 3-(b)) whereby this node is added to the pool 2 rather than pool 1 to which it belongs. This is to avoid the root node hogging one-third of the slots in its neighborhood for all node-densities. Hence for the network in Fig. 3-(a), node 1 contends with node 2 for slot 2. The *Membership slot* part of the algorithm (lines 2,3) is explained in III-C.

Algorithm 2 Transmission Algorithm

```

Let  $thn_i$  denote two-hop-neighborhood of node  $i$ .
if  $Time.slot.num \% Membership.slot.num == 0$  then
    State = Receive
else
    Pool =  $Time.slot.num \% 9 + 1$ .
    Calculate  $LabelA_i \% 3$  and  $LabelB_i \% 3$ .
    if Node  $i$  belongs to pool then
        Get contending nodes  $C_i$  in  $thn_i$  for pool.
        for all  $n$  in  $C_i \cup i$  do
             $Priority_n = Hash(n \oplus Time.slot.number)$ 
        end for
        if  $Priority_i > Priority_n$  for all  $n$  in  $C_i$  then
            State = Transmit
        else
            State = Receive
        end if
    else if  $LabelA_i \% 3$  is same for that pool then
        if No node in  $thn_i$  has same  $LabelA_i \% 3$  value then
            State = Transmit
        else
            State = Receive
        end if
    else
        State = Receive
    end if
end if

```

For a given pool, the transmission algorithm uses a hash function to calculate priority values for all contending nodes for that particular slot. A node wins the slot if it has the highest priority value for that slot. The hash function takes as input the *Slotnumber* and the *NodeID*. Since the slot number is the same across contending nodes that are synchronized, the hash function provides consistent priority values thereby avoiding collision between the nodes, and can also be modified depend-

ing on network characteristics and service requirements.

Assigning co-ordinates based on the distance of the node from a particular root can lead to the counting to Infinity problem present in the distributed Bellman-Ford routing algorithm. This is avoided in VCMA by using *root sequence numbers* and *Root-A* node identifier attached to all labels. The *Root-A* node increases the sequence number for every *Hello* that it sends. A node labels itself with a new label in relation to a particular neighbor node only if the neighbor has a smaller *Root-A* node-ID or if it has the same *Root-A* node-ID and a higher *root sequence number*. This condition also enables a network to maintain their virtual co-ordinates when the current *Root-A* node fails. Therefore, every time the *Root-A* node fails or when the network gets partitioned, global communication to establish new *Label-A* labels does not occur and scheduling continues without incurring any overhead in bandwidth and time. This makes the protocol highly tolerant to node failures.

C. Neighbor Discovery

Fig. 4. Transmission and Membership frames

The collision-free transmission scheduling in VCMA depends on accurate two-hop neighborhood information. It is critical for a node to realize and incorporate neighborhood changes promptly. The mechanism involved in dealing with topology changes is described briefly.

New members to channel access scheduling use the *Membership Frame* to transmit their *Hello*s and let the neighbors know of their presence. Nodes are differentiated in terms of those that are already participating in the channel access

scheduling and those that are not. The former nodes are silent during the membership section, when latter nodes announce their existence using *Hello*s.

A new member first listens to the network traffic for at least one complete *block* before it tries to participate in the scheduling, where a *block* consists of F consecutive frames. The parts are depicted in Figure 4. The duration of a block, which is F frames, is derived such that it is highly probable that every two-hop neighbor of the new member transmits at least once in the block. We consider two-hop instead of one-hop neighbors because the probability of each node being activated is the reciprocal of the number of its two-hop neighbors in a uniformly distributed network. This situation is same as the coupon collector's problem. Given the average two-hop neighborhood size n in a uniformly distributed network, the number of frames in a block such that the probability of all nodes in the two-hop neighborhood transmitting in the block is greater than 0.99 can be derived from the relation

$$P(t \geq cnH_n) \leq \frac{1}{c}$$

where t is the total number of slots taken and c is any constant value. The value of F would also depend on the expected mobility, if any, in the network. We have an upper and a lower bound on the block size such that less time is spent on neighbor coordination while new members still can quickly notify the network.

The new members detect the slot numbers from listening to the network on joining the network. When the pre-determined membership slot numbers come, the new member randomly selects a slot within the section to transmit its signal, which contains its ID number, label and those of its one-hop neighbors. It is expected that all one-hop neighbors of the new member hear the signal and incorporate the new member in their one-hop neighbor set.

IV. SIMULATION RESULTS

We present simulation results comparing VCMA with IEEE 802.11 DCF, FPRP and NAMA. All the protocols use the 802.11b physical layer with transmission rate of 11Mbps. We use goodput and channel access delay as our performance metrics. The goodput is the number of useful data bytes sent across the transmission link between nodes measured at higher layers in the protocol stack. This is to avoid counting the control packets and retransmitted data packets that might occur in throughput calculations at the MAC. The channel access delay is an indication of the fairness and the coupon-collector's problem in the protocols. The simulation was done for different terrain dimensions in order to change node density, neighborhood size and number of collision domains. The nodes have a transmission range of 250m. Mobility was not considered for the simulations.

We used the discrete event simulator Qualnet [20] version 4.5, which provides a realistic simulation of the physical layer, and a well-tuned version of IEEE802.11DCF. Each simulation was run for ten different seed values for a uniformly distributed

network of size 100 nodes. The slot duration for FPRP, NAMA and VCMA was set to 1 ms, with all protocols capable of transmitting multiple data or control packets during the same slot. The *Hello* packets are sent for both protocols at intervals of 500 milliseconds.

A. Goodput

In this experiment, the traffic load of the network is increased continuously. Unicast data packets are delivered to a one-hop neighbor that is chosen at random, and the average number of packets received in the network is measured at the application layer. The traffic pattern is used to find the maximum capacity of the network. The experiment was done for four different terrain dimensions, and hence four different node densities. It can be seen from Figure 5 that as the traffic load is increased, VCMA outperforms NAMA, FPRP and 802.11.

B. Mean and Standard Deviation of access delay

Access delay is the time delay between successive instants of time at which a node accesses the media for transmission. In this experiment, the average access delay is compared for 802.11, NAMA, FPRP and VCMA for different node densities. The mean access delay for VCMA is better than that of the other protocols, and as expected the mean access delay is inversely proportional to the packet delivery as expected. The standard deviation of the access delay for VCMA is lower compared to NAMA and 802.11. This is because lesser number of contending nodes result in nodes accessing the media at more regular intervals than in NAMA and 802.11. However, the standard deviation for FPRP is better than VCMA, because nodes have fixed time slots in the frame for transmission. It is clear from the data that the protocol reduces the coupon collectors problem, allowing nodes to access the channel more frequently and at more regular intervals.

V. CONCLUSION

We have introduced VCMA, the first MAC protocol for wireless mesh networks that establishes transmission schedules in a distributed manner based on virtual co-ordinates assigned to nodes that reflect their locations relative to neighboring nodes, rather than their own identifiers as in traditional MAC protocols. VCMA establishes transmission schedules without multiple rounds of negotiations among nodes, which is the case in many schedule-based MAC protocols based on reservations. It was shown through simulations that VCMA outperforms 802.11, FPRP and NAMA in terms of goodput, and also provides better channel access delays. Our work opens up a new area of research in distributed transmission scheduling based on the use of location-based co-ordinates that reflect the topology of the network, rather than just node identifiers.

REFERENCES

- [1] B.P. Crow, I. Widja, L.G. Kim, and P.T. Sakai. "IEEE 802.11 Wireless Local Area Networks". *IEEE Communications Magazine*, 35(9):11626, Sept 1997.

a) Goodput

b) Access Delay

Fig. 5. Goodput and Access Delay

- [2] S. Vasudevan, D. Towsley, and D. Goeckel, "Neighbor Discovery in Wireless networks and The Coupon Collector's Problem," *Proc. ACM MobiCom 2009*.
- [3] L. Bao and J.J. Garcia-Luna-Aceves, "A New Approach to Channel Access Scheduling for Ad Hoc Networks," *Proc. ACM MobiCom 2001—Seventh Annual International Conference on Mobile Computing and Networking*, July 16-21, 2001, Rome, Italy.
- [4] L. Bao and J.J. Garcia-Luna-Aceves, "Hybrid Channel Access Scheduling in Ad Hoc Networks," *Proc. IEEE ICNP 02: 10th IEEE International Conference on Network Protocols*, Paris, France, November 12-15, 2002.
- [5] J. Ju and V. Li, "An optimal topology-transparent scheduling method in multihop packet radio networks," *IEEE/ACM Transactions on Networking*, Vol. 6, No. 3, pp. 298-306, Jun. 1998.
- [6] Carlos H. Rentel, and Thomas Kunz, "On the Average-Throughput Performance of Code-based Scheduling Protocols for Wireless Ad Hoc Networks," *6th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, Urbana-Champaign, IL, May 2005.
- [7] I. Chlamtac, "Fair Algorithms for Maximal Link Activation in Multihop Radio Networks," *IEEE Transactions on Communications*, Vol. COM-35, no. 7, July, 1987.
- [8] C. D. Young, "USAP: A Unifying Dynamic Distributed Multichannel TDMA Slot Assignment Protocol," *Proc. IEEE MILCOM Oct '96*.
- [9] C. Zhu and M.S. Corson, "A Five-Phase Reservation Protocol (FFRP) for Mobile Ad Hoc Networks," *Wireless Networks*, Vol. 7, No. 4, pp. 371-384, Sep 2001.
- [10] Z. Tang and J.J. Garcia-Luna-Aceves, "Hop reservation Multiple Access for Multichannel Packet Radio Networks," *Computer Communications*, Elsevier, 2000.
- [11] R. Rozovsky and P. R. Kumar, "SEEDEX: A MAC protocol for ad hoc networks," *Proceedings of The ACM Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc 2001, pp. 67-75, Long Beach, Oct. 4-6, 2001.
- [12] L. Bao and J.J. Garcia-Luna-Aceves, "Distributed Channel Access Scheduling for Ad Hoc Networks," Chapter 4, *Handbook of Algorithms for Wireless Networking and Mobile Computing* (A. Boukerche, Ed.), Chapman and Hall/CRC, 2006.
- [13] L. Bui, S. Sanghavi, and R. Shirkant, "Distributed Link Scheduling with Constant Overhead," *IEEE/ACM Trans. Networking*, Vol. 17, No. 5, Oct. 2009.
- [14] C. Joo and N. Shroff, "Performance of Random Access Scheduling Schemes in Multi-hop Wireless Networks," *IEEE/ACM Transactions on Networking*, Vol. 17, No. 5, October 2009.
- [15] C. H. Rentel and T. Kunz, "A mutual network synchronization method for wireless ad hoc and sensor networks," *IEEE Transactions on Mobile Computing*, Vol. 7, No. 5, pages 633-646, May 2008.
- [16] P. Djukic and P. Mohapatra, "Soft-TDMAC: Software TDMA-based MAC over Commodity 802.11 hardware," *IEEE INFOCOM 2009*.
- [17] S. Giordano, I. Stojmenovic, L. Blazevic, "Position based routing algorithms for ad hoc networks: a taxonomy", in: X. Cheng, X. Huang, D.Z. Du (Eds.), *Ad Hoc Wireless Networking*, Kluwer, Dordrecht, 2003.
- [18] N. Wen and B. Berry, "Information propagation for location-based MAC protocols in vehicular networks," in *40th Annual Conference on Information Sciences and Systems (CISS)*, Princeton, NJ, March 2006.
- [19] S. Katragadda et al., "A decentralized location-based channel access protocol for inter-vehicle comm., in *Proc. of IEEE VTC 2003- Spring*, April 2003.
- [20] Qualnet4.5 <http://www.scalablenetworks.com>