Scalability Issues in Ad-Hoc Networks: Metrical Routing Versus Table Driven Routing

Yosi Ben-Asher, Sharoni Feldman and Moran Feldman
Computer Science Department, Haifa University, Israel
yosi@cs.haifa.ac.il, sharoni@cs.haifa.ac.il

Pini Gurfil
Faculty of Aerospace Engineering
Technion - Israel Institute of Technology, Haifa 32000, Israel
pgurfil@technion.ac.il

Abstract

Scalability in ad hoc networks is a problematic issue, most works presents experimental results for limited number of nodes (100-200) nodes in a field. Various "explicit" clustering techniques have been proposed to improve scalability obtaining successful sessions in fields of 400-800 nodes. However explicit clustering may damage the performances, e.g., sessions breaks due to fast movements of cluster heads and the overhead for the explicit partition to clusters. An alternative to explicit clustering is to use algorithms that are "naturally clustered", i.e., over time arrange the nodes in dynamic hierarchical structures obtaining a similar effect to that of explicit clustering. The explicit clustering is more adaptive than explicit clustering and basically comes without overhead as it does not require an additional protocol for explicit partition of the nodes to clusters and cluster heads. For example if a cluster head moves away from its group another node may replace it without updating its class member. In this work we study the effect of explicit clustering by comparing an advance version of the AODV (a core algorithm in ad hoc networks) with the MRA algorithm that has the naturally clustering property. We cover fundamental aspects of scalability and

experimentally prove the superiority of explicit clustering over explicit clustering. In particular we consider heterogeneous theaters with several types of transmitters including personal, cars, helicopters and a GEO satellite. Naturally clustering is more effective in heterogeneous theaters as the more powerful transmitters (helicopters) serve as cluster heads.

1. Introduction

Mobile ad-hoc networks (MANETs) are becoming increasingly attractive due to their instant deployment capability and independence of infrastructure. Ad-hoc networks constitute a natural solution for communication networks in a disaster zone where the fixed infrastructure is inoperative or in military applications where military forces must deploy in uninhabited areas. The ability of ad-hoc networks to preserve the connectivity among their members even when the participating nodes are moving has earned these networks with their reputation as *ubiquitous networks*.

One problem of existing ad-hoc protocols is scalability, namely increase the number of successful sessions proportionally to the underlying number of transmitters. For example, in [3] it is stated that "it has been proven that current routing protocols work well in small size networks (e.g. fewer than 100 nodes)". Ref. [9] presents the Extended HSR (EHSR) protocol with a network which consists of 100 nodes in a 1Km × 1Km theatre. Ref [10] makes calculations for a small number of nodes (8) and extrapolates the results for 100, 200 and 300 nodes. Most papers do not produce simulation results for more than 150 nodes and sessions. A simple argument can be used to show that for sufficient densities of nodes any ad hoc algorithm will be blocked, i.e., no successful sessions will be created. Hence, if the above bound is true then scalability in ad hoc networks is a matter of obtaining some constant number of sessions for sufficiently large densities and the number of successful sessions can not grow in proportion to the underlying number of nodes. Intuitively, this follows from the fact that in high densities packet lost due to queue overflow and MAC collisions is such that no packet is able to reach its destination.

The common routing method to increase scalability is to use hierarchical/clustered protocols such as the Hierarchical State Routing (HSR) protocol [4] or the Intelligent Hierarchical State Routing protocol (IHSR) [2]. These protocols divide the nodes in the spatial network into backbone nodes and regular nodes arranged in clusters. Every cluster uses a cluster *head node* that is a part of the backbone. The cluster head node acts as a local coordinator of transmissions within the cluster and is responsible for keeping and updating routing information beyond the cluster. Clustering requires that the cluster heads will be more powerful (transmission range and capacity) transmitters than the remaining nodes in each cluster. Typically, cluster heads may be Helicopters, UAVs or Cars with increased transmission range and capacity yielding a "heterogeneous theater" (see Figure 1). The use of clusters significantly reduces the traffic of packets in the underlying network as communication between nodes of different clusters is restricted to the cluster heads backbone. Quoting [2], "all these results show us that a homogeneous structure cannot be scalable to a large-size ad-hoc wireless network. Heterogeneous hierarchical structure should be the solution".

However, clustering too has several drawbacks: (1) there is a significant overhead to maintain the cluster (e.g., electing the cluster head and maintaining the cluster's members); (2) the centralization of routes via the cluster-heads [4], i.e., sessions that can be routed through two "near" clusters must now be routed through their cluster heads; (3) clustered protocols are more sensitive to breaks and faults of the cluster heads; (4) the number of cluster heads can be larger than the optimal number incurring significant overheads by creating too many small (5); the session path may require more nodes than a direct path; (6) leadership changes result in routing changes and hence generate routing overhead as session heads are relatively faster moving nodes.

In this work, we show that the newly developed Metrical Routing Algorithm (MRA) [6] based on virtual coordinates is capable of handling a heterogeneous theater and obtain high ad-hoc connectivity *without* using an explicit clustered backbone thus solving most of the above mentioned drawbacks of explicit clustering. The MRA is unique compare to most types of ad hoc networks algorithms as it does not build routing tables but maintains a dynamic set of coordinates to every node. Thus if the coordinates of the destination are known the MRA sends a message to this destination through the shortest path based on the estimated metrical distances. We mainly compare the MRA with an advance variant of the AODV protocol which is a core algorithm for ad hoc networks and most works in the field use it for comparison purposes. As explained earlier due to inherent bounds on ad hoc networks scalability in ad hoc networks is mainly subject to experimental validation. The experiments include the following aspects of scalability:

- Pure scalability measurements, i.e., number of successful sessions versus increasing number of nodes (same type). Two types of pure scalability are considered increased number of nodes with a fixed size field (implying increased densities) and scalabilty with fixed density of nodes.

1. Heterogeneous scalability, namely how well the underlying two algorithm scales up when using larger numbers of helicopters and cars. Obviously, for every combination of field size and number of nodes there is an optimal number of helicopters and cars maximizing the number of successful sessions.
2. Proving explicit clustering abilities of the MRA by showing that the MRA uses the more powerful nodes as cluster heads.
3. Measuring the "reverse effect" of explicit clustering wherein some sessions by-pass cluster heads.
4. Measuring the benefit of using artificial clustering for both the MRA and AODV. Here it turns out that as expected clustering is essential for the AODV's scalability but only limits the scalability of the MRA.
5. Relative effects of increased transmission range versus increased capacity.
6. Measuring the effect of the speed in which nodes are moving on the scalability.
7. The effect of using a GEO satellite, showing that the satellite has only minor contribution to the salability compare to helicopters and cars.

The main contribution of this work is in its detailed set of experiments measuring various scalability issues in a heterogeneous theater of ad hoc networks. The superiority of using natural clustering versus explicit clustering as is expressed in the MRA versus AODV experiments. Other relevant previous works have only considered very limited aspects of scalability in heterogonous ad hoc networks mainly the relationship between the designated backbone nodes and between the designated nodes and the ordinary nodes.

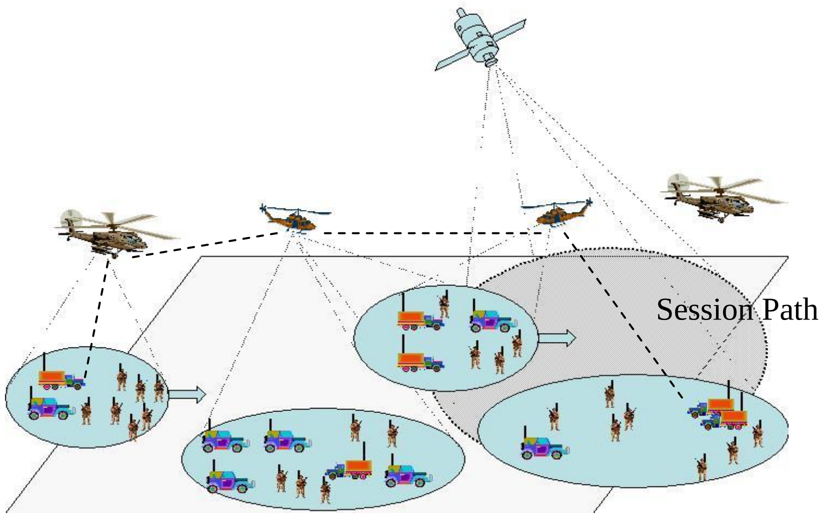


Figure 1: Heterogeneous Theater with an ad-hoc network layout

1.1. Background and related works

The background is organized as follows. We start by describing the AODV algorithm and its following extensions. Next we argue that it is sufficient to compare the MRA performances to those of the AODV as for our purposes other types of algorithms can be regarded as basically suffering from the same problems as those of the AODV. Finally we review proposed clustering techniques and discuss relevant scalability results.

Clustering for example was extremely helpful for the AODV algorithm which is a core algorithm for ad hoc networks. The AODV was proposed by Perkins and Royer [11]. The AODV is an on-demand protocol that floods the network with RTS messages whenever a node requires a path to a destination. Once discovered, a route is maintained as long as needed by source. The AODV creates the foundation for many variants and extension of the basic protocol like the Adaptive Routing using Clusters (ARC) [4] or MBNP-AODV[13] that utilizes the basic AODV with a clear distinction between backbone nodes, regular nodes and nodes that can become backbone nodes.

In this paper we divide the scope of ad hoc protocols into two fundamental families. The family of protocols that use geographical or virtual coordinates to find and maintain a path between the source and target nodes and the family that include the rest of routing protocols including proactive, reactive and hybrid. A significant difference between the two families is that the second family keeps and uses connectivity information like routing tables or other temporal data structures to keep routing information for rapid route establishment and quick recovery from path breaks. For example the FSR (Fisheye State Routing) [12] divides the space around a central node into two scopes. The adjacent scope is defined as a set of nodes that can be reached within a given number of hopes. Details about nodes within the adjacent scope are constantly and frequently propagated to the central node while information regarding the rest of the nodes is sent to the central node in a lower pace. As a result, a part of the routing information is kept very accurate while the other part is less accurate and more outdated.

The heterogeneous theater and the scaling issue are presented in the literature as two bounded issues. A large theater like a battlefield hosting a large number of heterogynous transmitters introduces the need to scale the network without compromising performance. There are many references in the literature to the scalability issues. However, no actual examples are given to establish the declarations that the proposed algorithms actually support scalability and the limits of the traffic load. Ref. [7] defines the scalability as "the ability of a network to adjust or maintain its performance when the number of the nodes increases". Ref. [2] describes a heterogeneous network where unmanned aerial vehicles (UAVs) are used to bridge between ground mobile entities. The Extended Hierarchical State Routing [EHSR] [9] protocol addresses the problem of routing in heterogynous networks with physically different networks at various levels. The algorithm differentiates between "ground backbone" and "aerial backbone". The dedication of specific nodes to serve as backbone nodes raises the question about situations where there is a lack of backbone nodes in some regions of the theater or crowding of backbone nodes in other regions.

The IHSR [2] is presented as a protocol that improves scalability by reducing the number of transmissions with the help of a-2nd level infrastructure. The simulations were performed on heterogeneous network with three types of radio interfaces. There is no description of scalability simulations and the results of such experiments. Ref. [1] discusses a heterogeneous network with ground nodes such as troops, ground mobile nodes and UAVs that maintain a line-of-sight connectivity. The discussion on scalability does not present any simulations of heterogeneous networks and performance analysis. Ref. [9] deals with a heterogynous network that can scale up but the simulations were performed with a fixed theatre of 100 nodes and a theatre size of 1Km × 1Km and a very slow and fixed speed of 2m/s of the backbone nodes. A simulation of a heterogeneous protocol and a comparison of clustering scheme with flat and hierarchical versions of AODV is presented in [3]. The mobile nodes speed was selected to be in the range 0-10m/s, with a large network of 1000 nodes. The nodes were equipped with two radios – one is like every ordinary node with limited transmission range and the other with extended range for backbone communications. The simulations do not include scalability tests. Ref. [5] reports on simulations performed to compare the H-LANMAR with flat LAMAR and flat AODV. The simulations include up to 36 backbone nodes with a single UAV connected to all backbone nodes in a theatre of 3.2Km × 3.2Km. No results are given on the scalability tests.

1.2. The upper bound for the number of successful sessions

General scalability in ad hoc networks implies that for a field with *n* nodes in a given density, and *I(n)* attempts to start sessions, the number of successful sessions *S(n)* is in proportion to *I(n)*, regardless of the scale of *n*, e.g., $S(n) = I(n)/8$. Below we show that even for modestly growing *I(n)* this aim is impossible and in fact with high probability almost all sessions will be blocked and will never reach their destinations. General scalability is as we will show possible only for ultra low values of *I(n)*. Let *c* be the capacity of every node, i.e., number of sessions that can pass through any given node. Clearly, for c = 1 one long session will block all other section to cross from one side of the field to the other, hence it is reasonable to require that the proposed bound will be valid for c > 1. Next one can always force the destinations of all sessions to be in a dense $\sqrt{I(n)} X \sqrt{I(n)}$ square allowing only $4\sqrt{I(n)} \cdot c$ sessions to reach their destinations. Thus, the proposed bound must be in a setting where potentially there is a way to rout all *I(n)* sessions to their destinations yet any "reasonable" algorithm will fail to find these routs.

Finally, Another reasonable requirement is that the bound will hold for $I(n) < c\sqrt{n}$ as there can be at most $c\sqrt{n}$ long sessions of length $\sqrt{n}$.

The following assumptions are used by the proposed bound:

1. There are *n* nodes randomly distributed in the plan such that they can be regarded as arranged in a mesh. The field size and the transmission range are selected such that with high probability each node has four neighbors. Alternatively, for any sufficiently large square with *m* nodes there are $\sqrt{m}$ nodes on every edge through which all sessions must enter or exit or do both.
2. Each node has a limited capacity allowing it to support at most *c* sessions. Control messages such as "*hello I am a neighbor*" are not charged.
3. If a node has allocated a bandwidth for a given session then its effective capacity *c* is decreased by one. When the capacity of a node reaches zero the node is blocked and will ignore/reject any messages of sessions other than those allocated.
4. Sessions are routed to their destination by allocation capacity in each node along some path to their destination.
5. For a given logical partition of the field to kXk squares (k > 3). we assume that each session must path through at least $k-2$ squares wherein due to the random selection of sources and destinations it must enters/leave through a random node on the external edges of each square it is passing through.

6. Let *s* be a session for which a path *p* has been allocated in a given square *q* such that there is "free capacity" on *p* for another session. We assume that if another session *s′* should enter and leave *q* at the same point as *s* then the algorithm will allocate the same path *p* to *s′*. This is the only questionable assumption we make on the way sessions are routed to their destinations. Intuitively we assume that the underline algorithm routs packets in an oblivious manner and it does not attempt to distribute loads. This assumption is common to all protocols we know. Intuitively a more "adaptive" algorithm attempting to preserve free capacities for "future" paths contradicts the dynamic distributed setting of current ad hoc algorithms.

7. A square is "blocking" if there are *c* sessions entering the square and leaving the square at the same nodes as depicted in Figure 2.

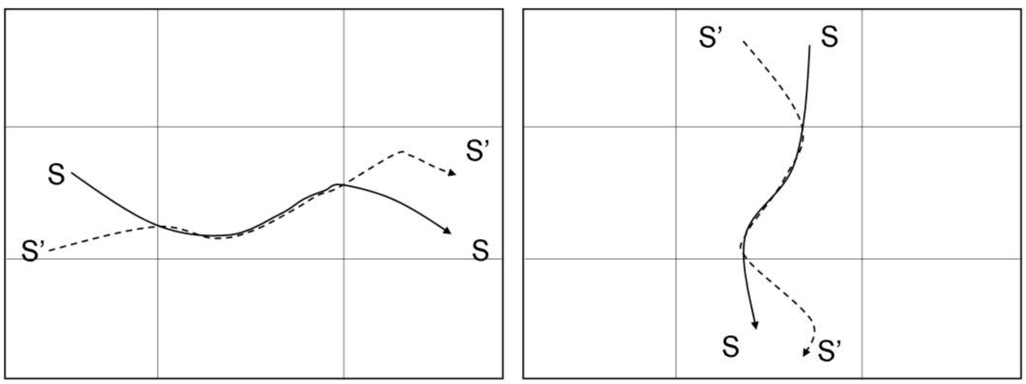


Figure 2 : Horizontal and Vertical Blocking

For complete blocking we need at least two such cases forming a horizontal and a vertical block (see Figure 3). We are assuming that if the squares are relatively small ($d < \sqrt{n}$) then the entrance point and the exit point of each session can be regarded as being selected at random. This assumption is supported by the fact that the sources and the destination of each session are selected at random and that modeling the exit point at each square as being selected at random can be regarded as a simplification of the complex situation that occurs when ad hoc routing is done in small squares. The probability of a dXd square with s sessions passing through to be blocked is $\binom{s}{c} \cdot \frac{1}{d^c}$.

For the proposed bound we will use c = 2, $s = \sqrt{\log(n)} \cdot \sqrt[8]{n}$ and $d = \sqrt[4]{n}$ such that the blocking probability is $\log(n)/\sqrt[4]{n}$.

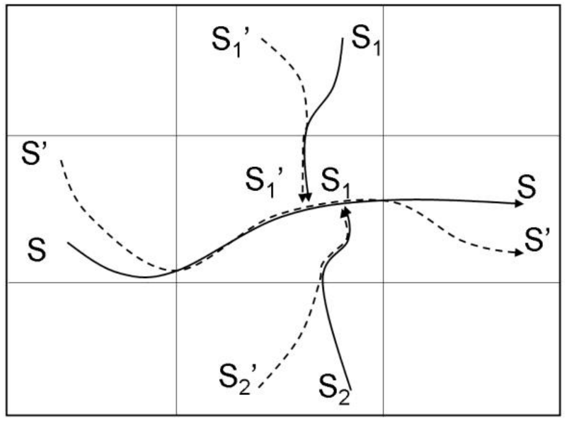


Figure 3: Blocking Square

8. A blocked session does not free the capacities allocated for it along the path until it reaches a blocking node. This is because we assume that there is a constant flow of requests for sessions and that freeing resources is a relatively a slow process leaving the allocated paths "live" for sufficient time to block other sessions.
9. The algorithm fails if for some partition of the plan to squares and any order in which paths are allocated to sessions in squares almost all sessions are blocked in some square.

For simplicity we will skip constants and o() notations neglecting only constants. We partition the field to $k = \sqrt{n}$ squares each of size $\sqrt[4]{n} X \sqrt[4]{n}$ nodes and set $I(n) = \log(n) \cdot \sqrt[4]{n}$ and c = 2. The sources and the destination are selected randomly such that each session must path through $\sqrt[4]{n}$ squares. The blocking probability (as computed earlier) of each square is $\log(n)/\sqrt[4]{n}$ hence the probability that a given session is not blocked in any of the $\sqrt[4]{n}$ squares it has to path is less then $\left(1 - \frac{\log(n)}{\sqrt[4]{n}}\right)^{\sqrt[4]{n}} \cong \frac{1}{n}$. Thus with high probability most of the *I(n)* sessions are blocked in some square. This bound hold for c > 2, however in this case $I(n) > 2^c \log(n) \cdot \sqrt[4]{n}$.

Note that this bound could be easily obtained for $I(n) = \sqrt[4]{n}$ had we selected the destinations of all sessions to be randomly distributed on the edges of an inner $\sqrt[4]{n} X \sqrt[4]{n}$

square as depicted in figure ?2. With high probability (as in this square $d = s = \sqrt[4]{n}$) this inner square will be a blocking square blocking most of the sessions. The fact that the bound holds for random selection of sessions across the whole field implies that this bound is also likely to happen in real life situations, i.e., this blocking effect should occur in actual simulations of ad hoc algorithms. The practical prediction of this bound is that for relatively large values of *n*, distance between any two nodes ≥ *r*, *c* = 2 and $I(n) = \sqrt[4]{n}$. If we increase *I(n)* there should be a decrease in *S(n)*. This novel prediction has not been observed before due to the relatively small values of n and other parameters usually used in ad hoc experiments. Degradation in *S(n)* due to high values of *I(n)* and high moving speeds is more likely to occur, the contribution of this bound is in showing that it can occur even for ultra low number of attempts to create sessions. The following results presented in figure ??? show that this phenomena occurs in actual simulations.

2. The Metrical Routing Algorithm

The Metrical Routing Algorithm (MRA) protocol [6] is classified as a virtual coordinates protocol, as some traffic control is used to maintain the mapping of the nodes. The MRA is capable of successfully handling a demanding traffic load under a high node density and fast node movement. The MRA organizes the nodes in rooted trees in order to find short session paths between nodes on the tree. The algorithm tries to minimize the number of trees by fusing separate adjacent trees into a single tree. As long as all nodes in one tree are not in the transmission range of all nodes in the other trees, the trees will function autonomously. As soon as a radio connection is created between two nodes, the trees will be fused into a single tree. All nodes run the same protocol implementing the MRA. As nodes emerge, disappear and move in or out of range of other nodes, there is need to update the trees. A primary goal of the algorithm is to identify these changes and adapt the trees structure to the new state. In the following discussion, we shall present an elaborate description of the MRA protocol, which will be ultimately employed for a simulation study of the MANET routing performance.

2.1 Dynamic Fusion of Spanning Trees

The MRA organizes the nodes in the field in rooted trees. Only nodes that belong to the same tree can create sessions among themselves. To ensure maximal connectivity, all nodes will try to organize themselves in a single tree. Every node in the field has a unique *node-id* (similar to a phone number or an IP address), and dynamic coordinates – the *node address* - that identify its location in the tree. Every tree is identified using a *tree name*, which is the *id* of the root node. Nodes periodically send beacons; every node that receives a beacon checks whether the node that sent the beacon belongs to a different tree. If the nodes belong to different trees, they will initiate a fusing process that will fuse the separate trees into a single tree. The fusion protocol should satisfy the follow properties:

1. The protocol should not cause active sessions to break.
2. Eventually (assuming no dynamic changes occur) all trees with nodes within transmission area must fuse into a single tree.
3. When two trees are being fused, most updates should be made to the nodes of the smaller tree (in terms of the number of nodes).
4. The protocol should maximize the number of nodes that migrate from one tree to another in every step (yielding a parallel fuse).
5. Nodes constantly attempt to shorten their distance to the root of the tree by fusing to higher level nodes.
6. Initially every node forms a separate tree of size 1.
7. The protocol is fully distributed with no central bottlenecks, namely it is defined at the level of pairs of nodes.

Every node in the tree can initiate a fusion process to a neighboring tree regardless of the node position in the tree. The fusion node gets new coordinates in its new tree according to the node's new position. Naturally, when a node migrates from one tree to a new tree, it may carry its neighboring nodes to follow it. Figure 4 presents three stages of the tree fusion protocol: The initial state, an intermediate state and final partition to trees (assuming no dynamic changes occur).

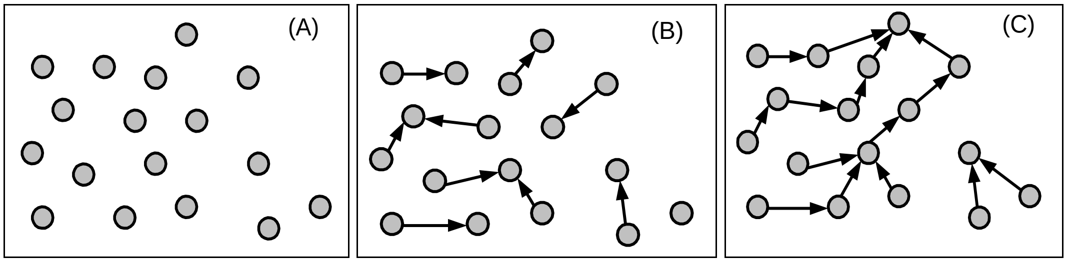


Figure 4: Tree formation process

Note that the two separate trees (C) cannot fuse because there are no two nodes within a transmission range that will start a fusion process.

In the following discussion, the terms "coordinates" and "address" are equivalent. The node address uniquely identifies the node in a tree. The children of every node are numbered, and the address of node *v* specifies the path from the tree root to *v*. For example, if the address of *v* is <0.1.1.2> then *v* is child number 2 of the node with the address <0.1.1>. A node can change its address during the fusing process when a node migrates from one tree to another tree or during maintenance processes of the tree. Figure 5 presents the addresses of the nodes in the trees and the address changes during the fusion process.

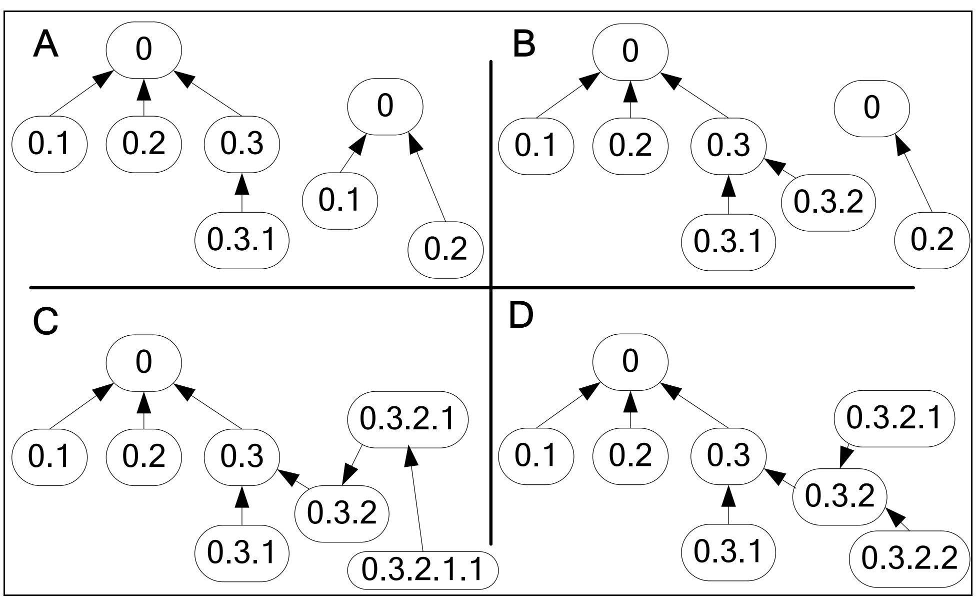


Figure 5: Tree fusion

Figure 6 presents a screenshot from the simulator, to be described in detail in Section 3, showing two separate trees fuse into a single tree.

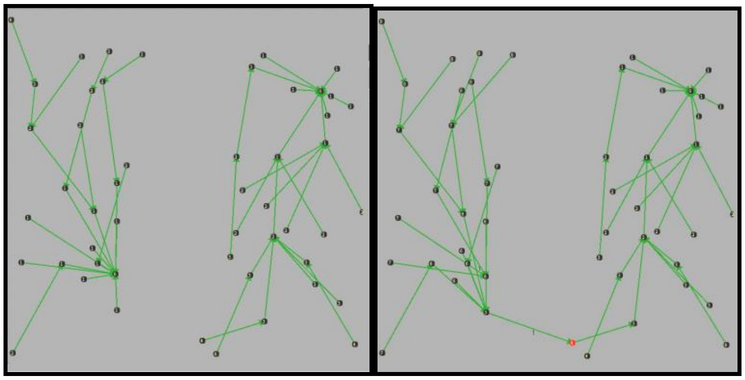


Figure 6: (a) **Two separate trees** **(b) Fused trees**

Continuing with definitions related to the MRA, *registration* of the node in the tree is the process of maintaining the mapping between the required *node-id* and the actual address is done in special *hash* nodes. The *hash* nodes are the children of the root (for example nodes <0.1>, <0.2> and <0.3> in Figure 5). Periodically, and after every migration/relocation, every node sends a registration message towards the root. This message contains the *node-id* and the *address* of the node. The registration is required to keep the updated address of the nodes identified by their *node-ids* in central locations in the tree (i.e. the root children). Un-refreshed entries in this table are aged and deleted.

2.2 Sessions

Every session occupies a caller, a called and if necessary, transit nodes. The caller is the proactive side initiating the session and the called and the transit nodes react to the session initiation request. The caller is in charge of resuming the session after a session break.

Called Resolution is triggered when a node *v* initiates a session to node *w*. Node *v* will interrogate the hash nodes in order to get the current address of *w* in the tree. For example, if node *node-id*=32 in Figure 7 tries to connect node *node-id* =602, it will get the address of node 602 from the hash nodes 0.1 or 0.2 or 0.3.

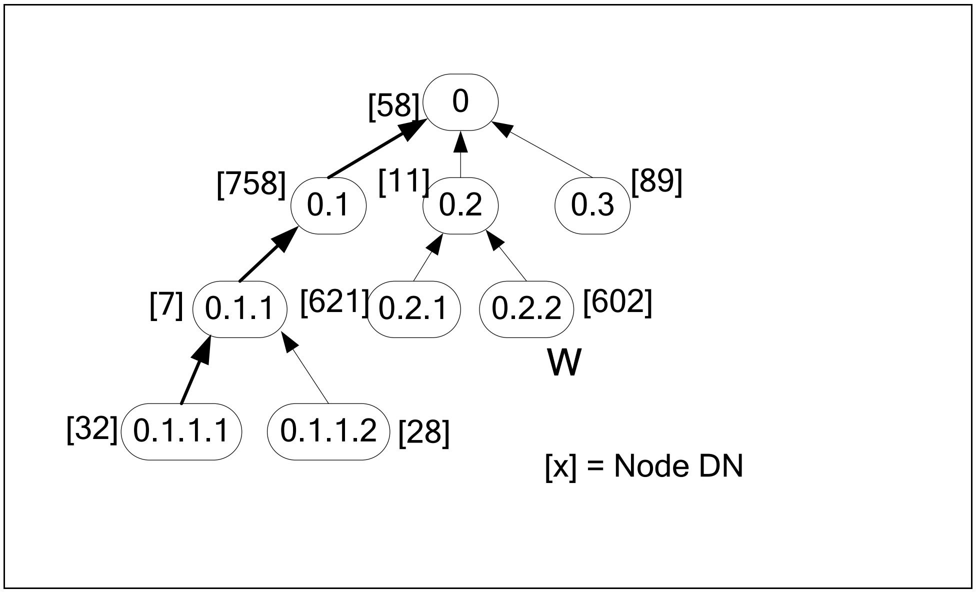


Figure 7: Address resolution

The *path allocation* procedure starts immediately after the caller gets the called address. This process creates the session path from the caller to the called. This process can fail because of various reasons, such as a busy called and unavailable resources to transfer the session. A *packets transfer* mechanism is utilized as soon as the path allocation process succeeded to create a path between the parties. The path is kept even if nodes along it change addresses or trees.

Session breaks occur if a node goes out of service or moves out of range. If a node notices that one of the adjacent path nodes does not transmit speech packets for a while, it assumes that the path has been disconnected and it clears the call from its tables. When the caller notices that the path is disconnected, it initiates a new path-finding process.

3. The IFAS: Interactive Flexible Ad Hoc Simulator

The IFAS simulator for evaluating the performance of the MRA and additional ad hoc protocols like AODV has been developed. In this section, we shall describe the simulator and the simulation scenarios.

3.1 Simulator Description

The simulator was designed and developed for testing the MRA and running comparative tests, comparing the MRA's performance to other routing protocols. Special attention was given to the following aspects: (i) enhanced visualization tools that give a full online view of the theater, node movements, voice channels, and specific node status including queue status; (ii) tracing the formation of trees in the MRA protocol; (iii) tracing the sessions in real time; (iv) configuration and simulation definition via online screens; and (v) support of logging, debugging and analysis tools.

The enhanced visualization capabilities, unique to this simulator, contributed to the understanding of the protocol behavior, as we were able to view the progress in the field and detect unexpected behavior. The simulator currently supports the following elements:

1. Parameter definition - the parameters are divided into two groups: *global parameters* and *protocol-dependent* parameters.
2. Field designer, enabling the user to enter obstacles in the field such as buildings.
3. Element definition tools.
4. Group definition with group mobility capabilities. Groups are managed autonomously.
5. Scenario loader, enabling the user to run recorded scenarios with different parameters and different protocols.
6. Field viewer, supporting the following layers: *trees view*, *session view*, *sparse trees view*, and *single node view*, including queues and queue content.
7. Offline analysis of the event logs, created during the test runs.
8. Parameter management tools.

Figure 8 presents the main parameter screen of the simulator. The main parameters groups in this panel are traffic parameters and operational parameters. The simulator enables the user to get detailed online reports on a single node behavior while the system runs. These capabilities set afloat disruptions in specific nodes behavior as a result of their location in the field. Figure 9 presents the entities management screen. It enables the user to define any number of entities in the field and control their behavior.

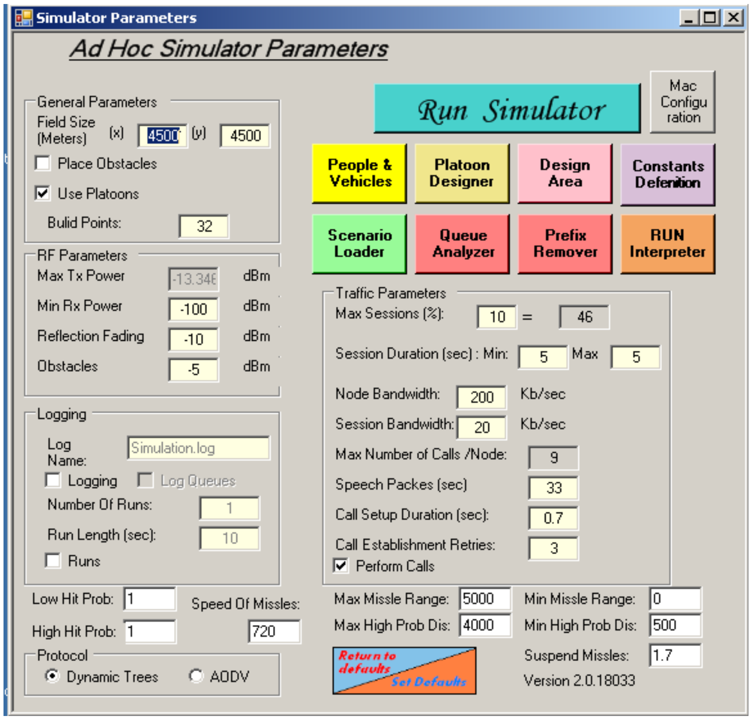


Figure 8: Ad-hoc parameters main screen

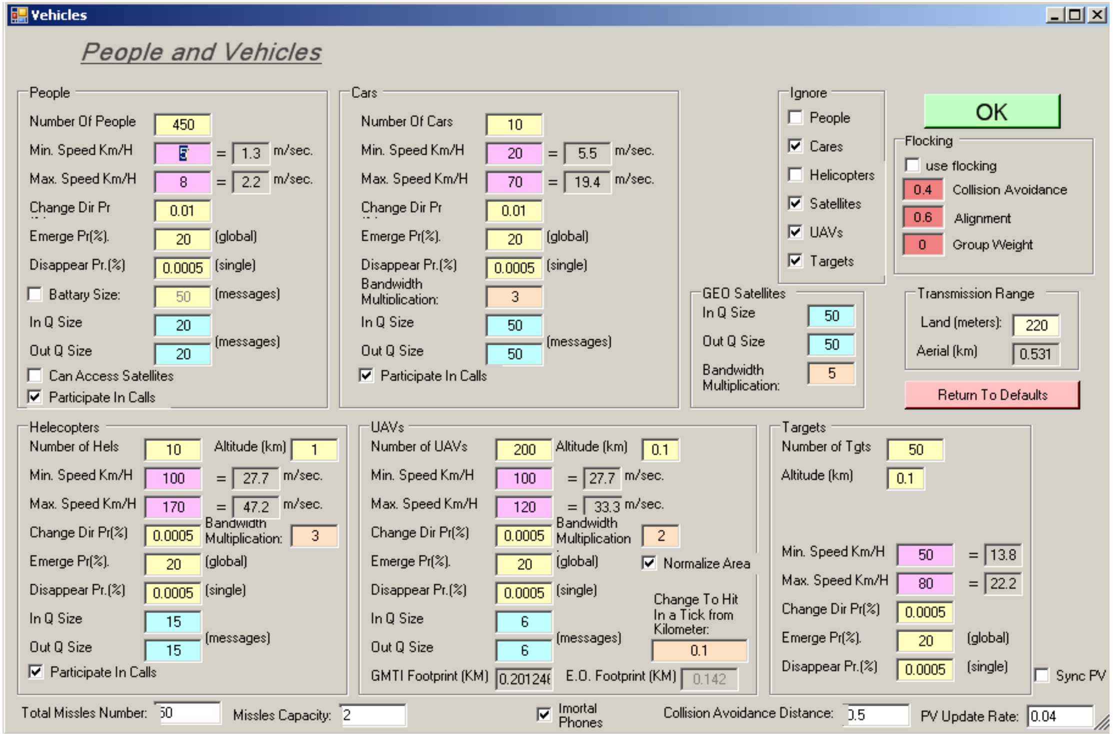


Figure 9: People and vehicle definition

3.2 Simulation Environment and models

We adjusted the simulation environment and the models parameters to every test. For example, simulations that involve satellites require a large scale field of the that can reach the size pf 60 km × 60 km while simulations that use only personal transmitters with short transmission range were performed in a small theatre of the size of 3.5Km × 3.5 Km. The node transmission range depended upon the following attributes: (i) transmitter type – a personal radio is limited by its transmission range and its battery power; (ii) the transmitter altitude - a transmitter mounted on a helicopter has a larger range than the same transmitter on the ground.

A session is a full duplex connection between nodes. When needed, one or more intermediate nodes will help to bridge the distance between the end nodes. A message can be lost because of an overflow of the queue in one of the chain of nodes used by the session or due to congestion due on the MAC level. Session duration is 5-15 seconds. The maximum session setup time is limited to 0.7 seconds and up to 3 recovery retries until a broken session is dropped.

Every simulation second is constructed of 330 ticks. The status of each node is evaluated every tick and decisions are taken. When using a satellite in a session, the transmission packets were delayed by 0.24 seconds. As opposed to other nodes, the satellite cannot generate or terminate sessions.

Table 1 depicts global parameters common to most simulations.

Attributes	Selected Values
Node Bandwidth	200 Kb/Sec. A single session bandwidth is 20 Kb/Sec.
Number of channels per node	Person: 10 channels with the following distribution: 1 channel dedicated for signaling, 9 channels dedicated for traffic. Car: up to 50 channels with the following distribution: 1 channel dedicated for signaling, rest channels are dedicated for traffic. Helicopter: up to 50 channels with the following distribution: 1 channel dedicated for signaling, rest channels dedicated for traffic. Satellite: up to 50 channels with the following distribution: 1 channel dedicated for signaling, 49 channels dedicated for traffic.
Number of messages/sec.	33 messages per session/second.
Number of Parallel sessions/calls generated by simulator	Aspire to 40% of the number of nodes (for example, 50 nodes – 20 parallel sessions, 60 nodes – 24 parallel sessions etc.)

Nodes insertion and removal (except satellite)	A new node will be inserted with the probability of 0.2 (per tick) as long as the total number of nodes in the area has not reached the maximum. Every node from the nodes in the field can be removed with the probability of 0.005 (per tick)
Node Queue size	Ajustable:, Satellite: In/out-Queue: 50 messages; Persons, Cars, Helicopters: In-Queue: 20 messages; Out-Queue: 20 messages

Table 1: Simulator parameter values

Mac Model

We implemented the IEEE 802.11 standard [8] for MAC of the nodes to accomplish wireless communications. This standard is the most widely deployed Wireless-LAN protocol.

Carrier sensing is accomplished through the use of Request-To-Send (RTS) and Clear-To-Send (CTS) control packets. Neighbors of the source and destination nodes receive the RTS/CTS packets and defer packet transmission in order to avoid collisions.

Every node has an inbound queue and an outbound queue. Every message generated by the node's applicative levels is stored in the outbound queue. Using FIFO policy, the node's MAC tries within every clock tick to send the oldest message from the queue. Messages trying to enter a full queue will be lost. A similar inbound queue exists for incoming messages. The analysis of the queues enables us to observe exceptional and unexpected behavior. The queues size can be adjusted according to the node type and the expected load (the load is composed of transit traffic where the node acts as a router, and traffic generated by the node when the node acts as a session originating node or session end node).

RF Model

The RF model is used to describe the radio propagation between any two nodes in the theater. The RF model is controlled by the Simulator control screens. It uses the Effective Radiation Power (ERP) formulas:

1. RF propagation free space transmission between two points at distance *d* is given by: $Loss = 92.5 + 20 \times \log(d \times f)$ Where *d* is in Km and *f* is GHz
2. RF propagation for antennas that are near the ground, the loss between a transmitting node and a receiving node is defined by the following formula: $loss = 40 \times \log(d) - 20 \times \log(Ht \times Hr)$ Where *d* is distance between antenna in meters and Ht and Hr is height of tower and height of receiver in meters.

3. Any physical obstacle increases the loss by 5dB.

Similarly to the management of the RF model, it is possible to manage the links bandwidth, packets transmission rate etc.

Nodes Movement model

A node (except satellite) starts to move with a random speed selected from the relevant range according to the node type. The node moves in this direction for a certain period of time until it changes its direction. A node that reaches the border of the theatre will be "reflected" back into the field.

Groups and Group Mobility

The group mobility supports relationship among mobile nodes. It allows the users to drag and drop "groups" in the theatre with dedicated movement capabilities. While the groups move in the theater according to a set of rules, the mobile nodes inside a group move according to the node nature. The tests designers are able to define what type and the number of nodes will populate every group. Every group has a mobility model of its own. The usage of groups is important when the theatre is very large and the nodes are arranged in units or when we like to force an entity to behave as an Artificial Cluster Head (ACL).

Logs

All simulation events are recorded into log files. The analysis is performed using offline utilities. These tools enable the designers to extract the relevant information from the logs.

4. Simulations and Results

The experiments that have been made were designed to verify the following claims regarding efficient ad hoc communication in a heterogeneous theater[1]:

- The MRA algorithm is naturally clustered so that hierarchical clustering is created without using clustering techniques. Unlike the AODV algorithm it does not require

[1] A theater with several types of transmitters with different moving characteristics, capacities and transmission ranges.

clustering to handle a heterogeneous theater. Thus the MRA scales better than the AODV in any combination of a heterogeneous theater.

- The MRA algorithm does not require clustering to handle increasing numbers of transmitters (of any type) and unlike the AODV it achieves scalability without using clustering.
- Using the "flat" version of the MRA instead of explicit clustering improves communication as the resulting clusters are connected not only through cluster heads (the most powerful transmitters) but also through personal transmitters (the weakest transmitters).

In all the experiments we are interesting to compare absolute numbers of successful sessions and also relative factors in which the number of successful sessions changes. We focus only on realistic scenarios that may reflect potential situations in a large theater.

Experiment set 1 – Scalability in the number of nodes

The scalability of the MRA compare to that of the AODV has been verified in a sequence of tests where the number of nodes in a fixed theater increases and the number of successful session is measured. The results show that for the MRA the number of successful sessions remain proportional to the number of nodes, hence MRA scales well. The AODV on the other hand was shown to scale down when the number of nodes increased. These results repeat itself for any combination of persons, cars and helicopters, in fact heterogeneity for the MRA seams to increase its scalability.

The scalability depends on the quality of sessions, and with quality of 100% scalability of the MRA drops but remains >1 (compare to the AODV's scalability that at 100% hardly exists). Finally, for a given number of persons there are always an optimal number of helicopters and cars that maximize the number of successful sessions. Such optimal combinations are the results of the negative effect of increasing the number of strong transmitters (cars and helicopters). This is because adding too many helicopters and cars increase queue's overflow and packet lost.

Table 2 presents in the left side of every cell the number of successful sessions for increasing numbers of personal transmitters (100-380) where the size of theater remains fixed (yielding increased densities). Note that the size of the message queue in every node remains fixed for all the experiments. The right side of every cell is the same but includes 10 helicopters and 10 cars in each experiment. The performance is measured by the number of sessions through which more than a certain percentage of the packets passed. The results shows that while the MRA scales up linearly for increasing numbers of personal transmitters the AODV's performances scales down with more than a quadratic factor. This result is true for both the uniform case and the heterogeneous case. The same relation between the AODV and MRA holds for when the results on the horizontal direction of the table. A special attention should be given to the case of 100% success when the AODV collapses due to its inability to manage successfully the extreme number of control messages. As indicated before every ad hoc protocol is subject to a fall in the number of successful sessions when the density pass a certain threshold. In this experiment the MRA's performances starts to drop when the density becomes greater than 40 transmitters per square KM.

Protocol	No. of Nodes	Success Rate				
		80%	**85%**	**90%**	**95%**	**100%**
MRA	100	270 / 380	267 /366	264 / 311	257 / 290	71 / 157
	140	341 / 401	325 / 388	315 / 354	289 / 321	67 / 112
	180	477 / 530	462 / 504	444 / 477	410 / 432	95 / 188
	220	524 / 619	489 / 583	452 / 521	383 / 407	95 / 407
	260	535 / 653	491 / 617	441 / 563	377 / 443	94 / 209
	300	604 / 690	558 / 632	518 / 596	442 / 485	112 / 187
	340	480 / 532	315 / 429	210 / 263	127 / 211	18 / 104
	380	178 / 203	131 / 168	75 / 102	29 / 64	7 / 17
	420	58 / 71	52/ 61	35 / 43	11 / 13	2 / 3
	460	8 / 9	5 / 6	2 / 2	1 / 1	0 / 2
AODV	100	188 / 231	188 / 207	188 / 196	184 / 194	45 / 68
	140	210 / 256	210 / 236	210 / 227	197 / 119	25 / 32
	180	252 / 268	226 / 239	172 / 181	90 / 112	6 / 24
	220	176 / 286	134 / 201	81 / 143	31 / 56	0 / 14
	260	40 / 73	23 / 65	12 / 48	4 / 23	0 / 14
	300	13 / 36	7 / 26	3 / 22	0 / 11	0 / 4
	340	2 / 2	0 / 8	0 / 4	--	--

Table 2: Persons successful sessions: MRA vs. AODV

It is interesting to consider the "absolute scalability" behavior of the MRA where both the number of nodes and the theater size increases so that the density remains fixed. Figure 10 Presents the results of scalability tests of the MRA with a networks growing from 100 to 2000 nodes with fixed density. Every node is capable to handle up to 10 parallel sessions. The results present a constant and linier growth in the network performance as there is no shortage of resources. A similar test with the AODV shows that the performance starts to drop much earlier. The absolute speedup of an ad hoc algorithm is a significant indicator to its quality as when the density remains fixed performances drops only due to instability of the ever growing length of the session paths. Figure 11 presents a view of a heterogynous theatre with 600 personal transmitters and 100 cars.

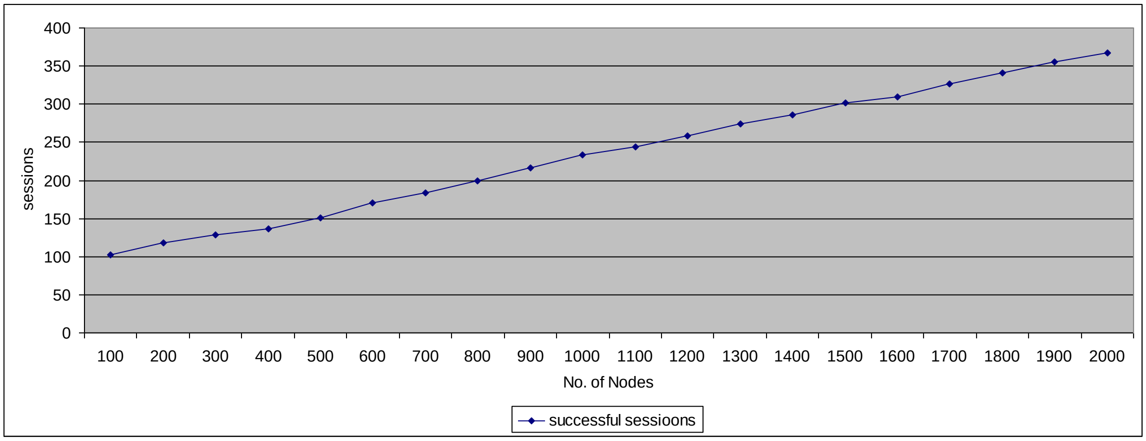


Figure 10: scalability of the MRA with fixed density

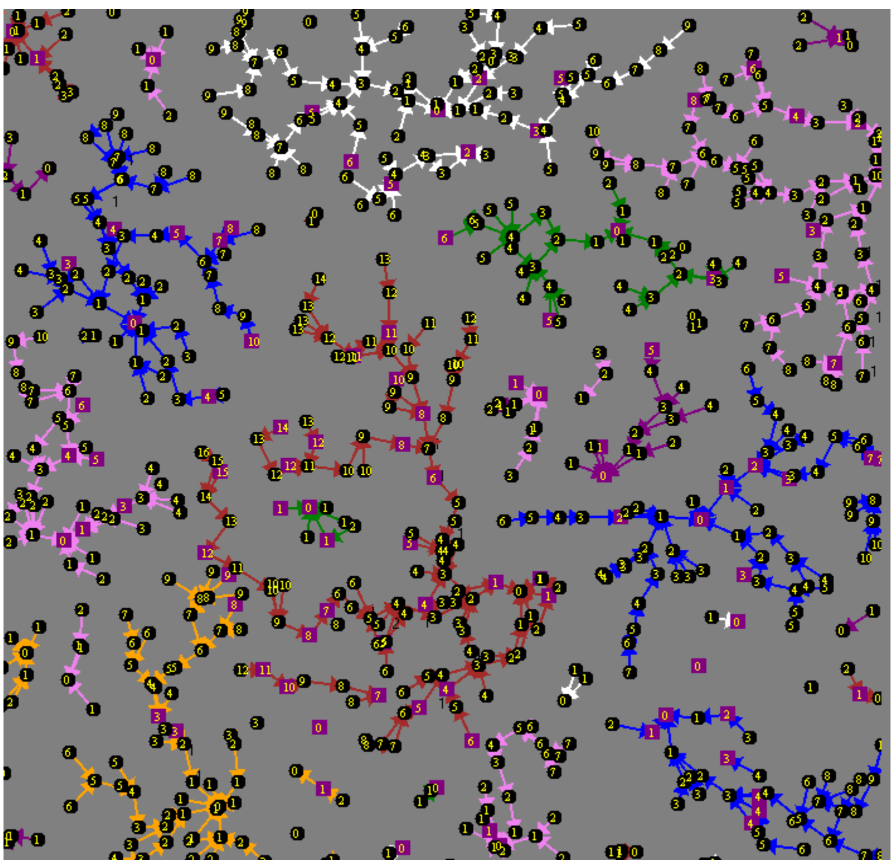


Figure 11: A heterogynous theater with 600 (circles) persons and 100 cars (squares)

Experiment set 2 – Explicit clustering

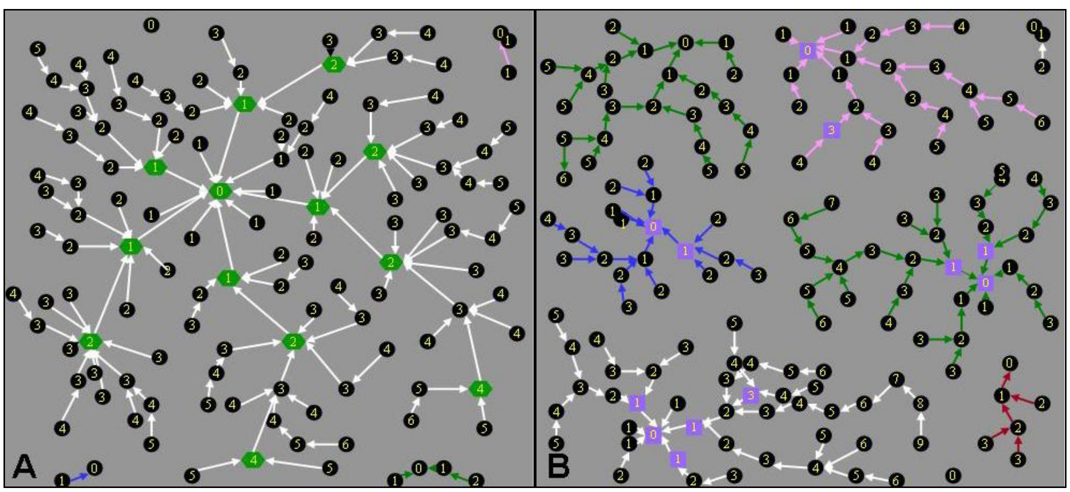
The explicit clustering property of the MRA is demonstrated using screenshots and log files. Basically we obtained screenshots showing that the stronger transmitters (cars and helicopters) migrate to the higher levels of MRA trees serving as cluster heads to the weaker transmitters, that are arranged in clusters corresponding to sub-trees. As for the AODV screenshots and log files do not reveal "explicit clustering" abilities as most session routes do not rely on stronger transmitters (candidates for cluster heads) to bridge large distances. Log files have been used to analyze clustering by classifying the communication routs of sessions. For example a session route of the form

person ⇔ *person* ⇔ *car* ⇔ *helicopter* ⇔ *helicopter* ⇔ *car* ⇔ *person* ⇔ *person*

is considered as an evidence to "explicit clustering" while a session route of the form

person ⇔ *helicopter* ⇔ *person* ⇔ *person* ⇔ *car* ⇔ *person*

is considered as an evidence of a lack of explicit clustering.



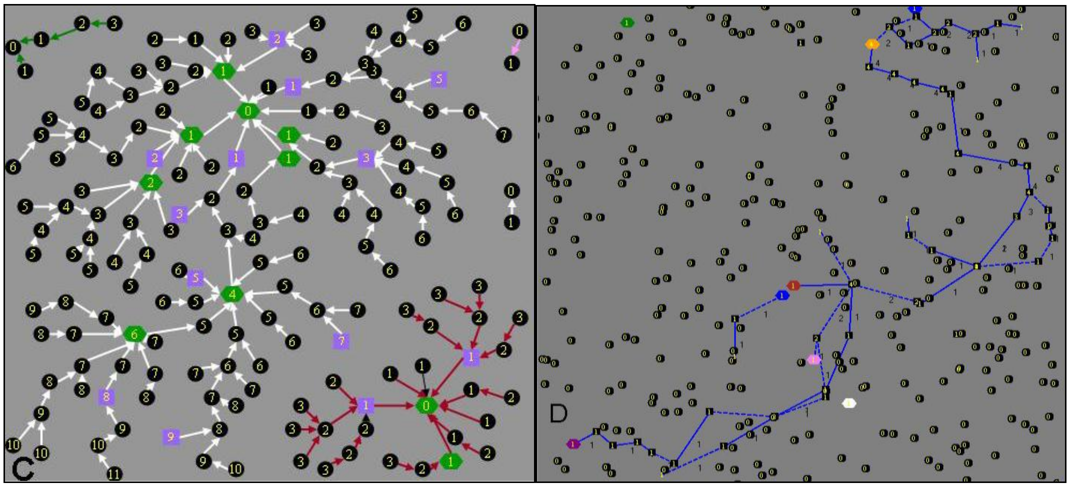


Figure 12: MRA trees and AODV screenshots

The heterogenic theater creates a natural hierarchy. Figure 12 presents snapshots that present the ability of the “stronger entities” to climb over time in the tree and position themselves in the upper parts of the tree created by the MRA.

Four random samples of screenshots are used to demonstrate the explicit clustering ability of the MRA:

- Figure 12-A depicts the fact that most of the helicopters (hexagons) are positioned in the upper levels of the tree. Note that the proportion between helicopters and personal transmitters is 1:10 but 40 percent of the possible nodes in levels 1 and 2 and the root are populated with helicopters. The long transmission range of the helicopters leverages its capability over personal transmitters to identify an unpopulated entry in a higher level and initiate the migration process.
- Figure 12-B presents the case when helicopters are replaced by cars whose transmission range is the same as personal transmitters only the capacity is tripled. The analysis of Figure 12-B shows that the cars crawled over time to higher levels on the tree. This is explained by the fact that the number of lost packets on a personal transmitter is higher than the number of lost packets on a car or helicopter transmitter as the latter bandwidth is much broader. As a result, fewer packets are lost. Part of the lost packets are control packets that handle the migration process. As a result, the chances that a personal transmitter that starts a migration process will complete it successfully are lower than the chances that a car mounted transmitter will complete it successfully.

- Figure 12-C presents the situation where the theatre hosts personal transmitters, helicopters and cars. It is visible that the helicopters crawl to higher positions on the tree before the cars and persons.
- Figure 12-D presents a snapshot of the theater using the AODV protocol. As can be seen the stronger entities take a weaker role as cluster heads.

The explicit clustering ability of the MRA is also verified by analyzing the sessions' type using detailed log files. Table 3 presents the percentage of clustered sessions (as defined earlier) that were created in a set of test simulations. The number of clustered sessions grows significantly as the number of hops (used to bridge the session end-nodes) grows. The usage of the cluster heads contributes also to the ability of the MRA to create shorter paths than the AODV as presented in Table 5.

Protocol	Percentage of clustered sessions				
	3 hops	4 hops	5 hops	6 hops	7 hops
MRA	48%	54%	59%	64%	75%
AODV	12%	14%	17%	21%	24%

Table 3: Percentage of Clustered Sessions from X hops Sessions

<u>Experiment set 3- Artificial Clustering Vs. Explicit clustering</u>

Another set of experiments demonstrating explicit clustering versus artificial clustering (groups) and measure its effect on the number of successful sessions. Note that artificial clustering forms an ideal clustering scenario as each group is separated from its neighboring groups hence communication between clusters is limited to cluster heads. It is shown that while artificial clustering does not help the MRA it is essential in order to lift up the performance of the AODV toward those of the MRA.

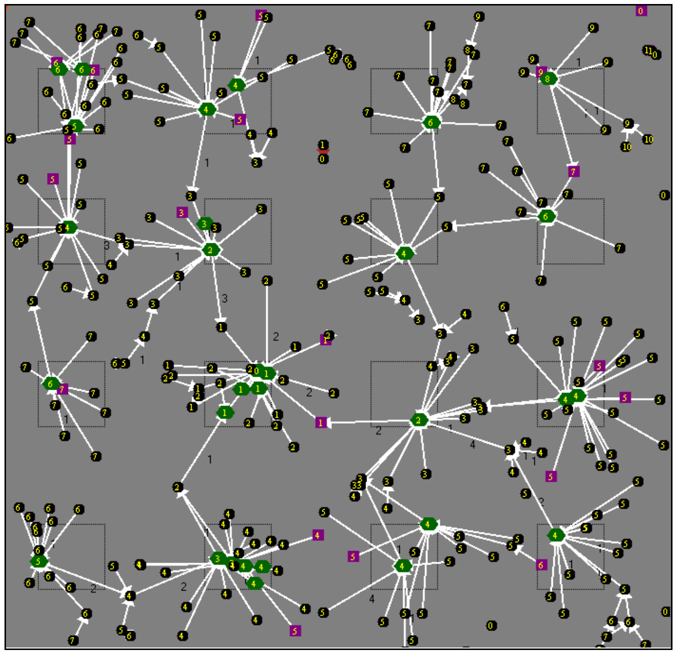


Figure 13: Artificial Clusters (ACLs) in Theater

Figure 13 presents the partition of the theater into 16 artificial clusters (ACLs). The theater hosts 300 persons, 20 cars and 50 helicopters. A helicopter/car/person that reaches the border of an ACL will turn back into the ACL. This construction forces powerful transmitters that can bridge two adjacent ACLs to serve every ACL.

Protocol	Clustering Type	
	No Clustering	Artificial Clustering (ACL)
AODV	118	425
MRA	930	673

Table 4: MRA and AODV - number of successful sessions with ACL and without ACL

Table 4 presents the results of test runs comparing the number of successful sessions generated by MRA and AODV with and without ACLs. The following insights result from our simulations: (a). The AODV performs much better in the clustered theatre. (b) The MRA performs much better in the case where no ACL forces helicopters to stay in a specific area. (c) The performance gap between MRA and AODV grows when no ACL is used.

Protocol	Clustering Type	
	No Clustering	Artificial Clustering (ACL)
AODV	6.7	6.2
MRA	5.6	5.8

Table 5: MRA and AODV average number of legs

Efficiency of an algorithm can be measured by measuring the average session path length allocated by every protocol. The results presented in Table 5 shows that the MRA obtain shorter sessions than the AODV. The findings presented in Table 5 correspond with the results presented in Table 4.

Cars	0	5	10	15	20	25
Sessions	22	23	25	28	30	31

Table 6: Successful sessions in heterogynous theatre with persons and cars

Helicopters	0	1	2	3	4	5	6
Sessions	22	84	206	250	320	325	330

Table 7: Successful sessions in heterogynous theatre with persons and Helicopters

Table 6 and Table 7 present the contribution of cars or helicopters to the connectivity. Table 6 presents the contribution of cars that have short transmission range and Table 7 present the contribution of helicopters with a long transmission range. The results show the significant contribution of the long range transmitters of the helicopters over the moderate contribution of the cars to the connectivity.

Experiment set 4- increased capacity versus increased transmission range

The following tests were targeted to analyze the contribution of additional bandwidth and transmission range. We divided the tests into 2 categories –Medium density theatre with 20-25 nodes per Km^2 and high density theatre with more then 25 nodes per Km^2. Basically in the medium range increasing the transmission range and bandwidth improves the number of successful sessions for both AODV and MRA (so both algorithms scale up), however the MRA scales approximately twice better than the AODV. Again this advantage of the MRA is explained by the explicit clustering ability moving helicopters and cars to the upper levels of the tree.

As for the high density case increasing the transmission range and bandwidth decreases the number of successful sessions. In high densities there is a "quadratic growth" most of the limiting factors of ad hoc algorithms including: number of control messages, queues overflow and MAC delays caused by overlapping of broadcasts. Note that in this case the AODV scales down faster than the MRA which, due to the explicit clustering, "absorb" better the quadratic growth effect. For example with 60 channels and all transmission ranges the MRA drops from 296 successful sessions to 282 while the AODV drops from 171 to 83 successful sessions.

Transmission Range (meters)		Cars and Helicopters Capacity		
Persons & cars	**Helicopters**	**10 channels**	**20 Channels**	**30 Channels**
235	500	45/32	72/45	110/71
250	550	98/66	115/83	144/99
265	600	135/84	153/92	199/113
280	650	142/115	185/125	242/137

Table 8: Medium Density –No. of successful sessions - Transmission Range Vs. link Capacity

Transmission Range (meters)		Cars and Helicopters Capacity		
Persons & cars	**Helicopters**	**40 channels**	**50 Channels**	**60 Channels**
300	700	270/132	295/168	296/171
315	750	266/120	306/131	309/130
330	800	250/113	260/121	265/124
345	850	233/94	275/85	282/83

Table 9:High Density –No. of successful sessions - Transmission Range Vs. link Capacity

Experiment set 5 – Speediness vs. Slowness

Another factor that was tested is the influence of the mobility characteristics on the clustering stability. It is shown that up to a certain level increased mobility has no significant impact on the number of successful sessions and clustering behavior. However, after a certain threshold, the performance drops significantly and the number of broken sessions increases significantly. The performance of the AODV protocol drops quicker than the MRA protocol showing that the AODV performs poorly under the extreme conditions. Note that in the case of ad hoc networks high mobility may break off active sessions. A broken session requires the session end nodes to initiate a recovery procedure aimed to reconnect the partners in a short period of time. Thus experimenting with high mobility is essential for testing the "heart" of ad hoc routing algorithms.

The experiments (whose parameters' ranges are given in Table 10) have been performed for two cases: artificial clustering (ACL) and explicit clustering (ECL) where no groups are used. The results as depicted in Table 11 presents the following insights: (1) regardless of the protocol used, the connectivity drops significantly as the speed grows and (2) Fhe average number of hops dropped significantly by 40%. This decrease indicates that the number of long session paths decreased due to the inability of the node to maintain the rapid changes. (3) For ECL the number of successful sessions of the MRA decreased by a factor of 2 compare to a decrease by a factor of 6 of the AODV. The decrease factor in the case of ACL is an average factor of 4 for the MRA and an average factor of 6 for the AODV. This demonstrates the superiority of explicit clustering and also the dependency of the AODV in clustering techniques to overcome high densities. Note that in average for both ACL and ECL the number of successful sessions was twice more than those of the AODV. Consequently the MRA improves upon AODV both in scaling abilities (decrease factor) and absolute numbers of successful sessions.

Entities	**No**	**Movement Speed**		
		Slow	**Medium**	**Fast**
Personal transmitters	300	1-2 Km/h	4-5 Km/h	8-10 Km/h
Cars	10	20-30 Km/h	45-55 Km/h	70-120 Km/h
Helicopters	10	20-30 Km/h	90-110 Km/h	170-230 Km/h

Table 10: Entities speeds

Protocol			**Number of successful sessions that succeeded with more than x%**				**Average session hops**
			80%	85%	90%	95%	
MRA	Slow Movement	ACL	128	121	104	98	6.1
		ECL	161	138	125	104	5.8
	Medium Movement	ACL	101	91	78	65	6.3
		ECL	138	107	98	83	5.9
	Fast Movements	ACL	40	40	29	18	5.5
		ECL	63	62	60	48	4.6
AODV	Slow Movement	ACL	84	78	65	44	6.7
		ECL	74	73	34	19	6.2
	Medium Movement	ACL	49	46	34	33	6.6
		ECL	43	38	31	17	6.2
	Fast Movements	ACL	15	15	11	7	3.8
		ECL	12	11	9	3	3.8

Table 11: Variable speed results

<u>Experiment set 6 – Reverse Effect</u>

In clustering algorithms communication sessions between cluster heads are solely made through cluster heads. This prevents the use of routing paths where cluster heads communicate via the less powerful transmitters. The MRA being naturally clusters does not enforce such a restriction. Consequently, as demonstrated before (Figure 13) the MRA functions better with ECL than when ACL is used.

We tested this “reverse effect” by evaluating the number of successful sessions between helicopters after adding different combinations of cars and Persons (Table 12). These tests present the ability of the low level elements to contribute in a very dispersed theater to the global connectivity.

Protocol	Entities in the theatre	Successful sessions	Average session hops
MRA	7 helicopters	24	2.3
	7 helicopters + 20 cars	33	3
	7 helicopters + 300 Persons	53	3.6
	7 helicopters + 20 cars + 300 persons	67	3.6
AODV	7 helicopters	25	2.3
	7 helicopters + 20 cars	32	3
	7 helicopters + 300 Persons	42	4.3
	7 helicopters + 20 cars + 300 persons	50	4.2

Table 12: Reversed hierarchy results

Experiment set 7 – the Contribution of the GEO Satellite

Finally, we tested the contribution of a GEO satellite to the field with personal transmitters, helicopters and supporting satellite. As presented in Figure 14 the GEO satellite covers the most of the theatre and creates a global cluster head. A helicopter can communicate directly personal transmitters or other helicopters when they are within transmission range. Another way to communicate between helicopters is via the satellite when the helicopters are within the footprint of the satellite. Note that a personal transmitter can connect only a personal transmitter or a helicopter.

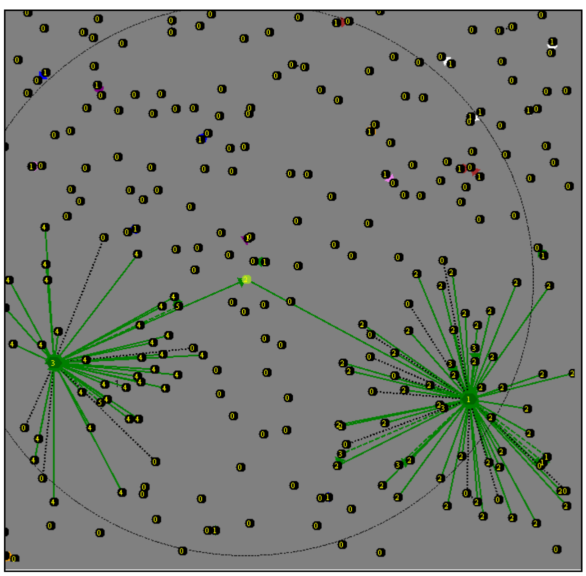


Figure 14: A theater covered by a satellite (in the center)

We tested 4 cases presented in Figure 15. In two similar tests we added gradually helicopters to a theatre with 300 personal transmitters. We measured the number of successful sessions created with and without the existence of a satellite. Both tests were executed using the MRA and AODV protocols. The results present the fact that contribution of the satellite to the total number of successful sessions is equivalent of adding one or two helicopters.

The contribution of the satellite is therefore not in its ability to increase the number of successful sessions but rather (as depicted in Figure 15) prevent the decrease in performances when more than the optimal number of helicopters are used. Indeed the lowering tails in the curves without the satellite disappear when the satellite is added. The decrease in the number of successful session when the number of helicopters grows over 8, results from the queues overflow and the inability of the MAC layer to transfer successfully all messages.

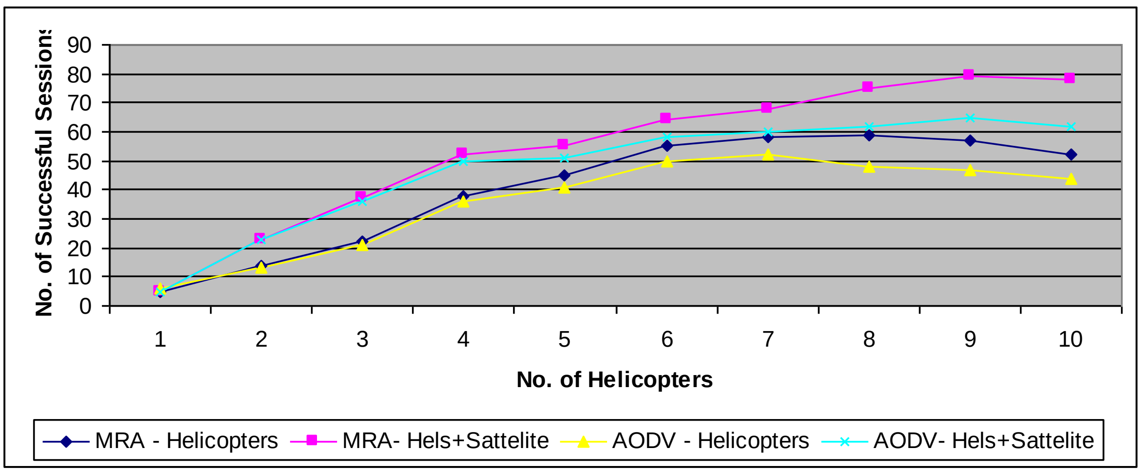


Figure 15: Substitution between Helicopters and Satellite

5. Conclusions

We described in this research a study of complex node and group behavior in an ad-hoc network constructed of heterogynous nodes. The research investigated the ability of the Metrical Routing Algorithm (MRA) to scale and maintain the connectivity between the nodes. The simulations performed with a realistic mobility model including personal transmitters, cars Helicopters and optional geostationary (GEO) satellite.

The main observation is that the MRA scales up and supports efficiently large networks constructed of heterogonous nodes. The ability of the MRA to create Natural Clusters (NCLs) by enabling the more powerful nodes to crawl to the upper levels of the tree and become cluster heads.

The helicopters provide high quality communication links. These links do not ensure a global coverage of the field. Additions of helicopters to the field broaden the covered area but do not guarantee a global coverage. A GEO satellite has a complimentary contribution to the coverage as by its ubiquity it contributes only to connect between cluster heads that cannot communicate directly or via other nodes in the theatre.

The "reverse scalability" tests show the ability of the MRA to create an efficient network in case that less powerful transmitters acts as NCLs.

References

[1] K. H. Rhee, Y. H. Park, G. Tsudik, “An Architecture for Key Management in Hierarchal Mobile Ad Hoc Networks”, Journal of Communications and Networks, vol 6, No. 2, June 2004.

[2] D. L. Gu et al., “UAV-Aided Intelligent Routing for Ad-Hoc Wireless Networks in Single-Area Theater,” In IEEE WCNC, 2000. http://citeseer.ist.psu.edu/gu00uav.html.

[3] K. Xu., M. Gerla, “A Heterogeneous Routing Protocol Based on a New Stable Clustering Scheme”, www.cs.ucla.edu/NRL/wireless/uploads/kxu_milcom02.pdf

[4] E. M. Blending-Royer, “Hierarchical Routing in Ad hoc Mobile Networks”, Wireless Communication & Mobile Computing, 2(5), pp. 515-532, 2002.

[5] K. Xu, X. et al., “Landmark Routing in Large Wireless Battlefield Networks Using UAVs, ” www.cs.ucla.edu/NRL/wireless/uploads/kxu_milcom01.pdf

[6] T. Ben-Asher, S. Feldman, M. Feldman, “Ad-Hoc Routing Using Virtual Coordinates Based on Rooted Trees” IEEE SUTC 2006, Taiwan.

[7] L. Villasenor-Gonzalez, Y. Ge, "HOLSR: A Hierarchical Proactive Routing Mechanism for Mobile Ad Hoc Networks," IEEE Communication Magazine, July 2005.

[8] IEEE Computer Society. IEEE 802.11 standard, IEEE standard for information technology

[9] D. Lihui Gu, G. Pei, H. Ly, M. Gerla, X. Hong, "Hierarchical Routing for Multi-Layer Ad Hoc Wireless Networks with UAVs", Proceedings of Milcom 2000.

[10] J. Li, Z. J. Hass, M. Sheng, "Capacity Evaluation of Multi-Channel Multi-Hop Ad Hoc Networks", TCPWC 2002

[11] C. E. Perkins, E. M. Royer, “Ad hoc on demand distance vector routing”, In proceedings of the 2nd IEEE workshop on mobile computing systems and applications, pages 90-100, New Orleans, LA, February 1999.

[12] A. Iwata, C. Chiang, G. Pei, M.Gerla, T. Chen, “Scalable Routing Strategies for Ad Hoc Wireless Networks”, IEEE journal on selected areas in communications, Vol. 17, No. 8, August 1999.

[13] I. Rubin, A. Behzad, H. Ju, R. Zhang, X. Huang, Y. Liu and R. Khalaf, "Ad Hoc Wireless Networks using Mobile Backbones," in Emerging Location Aware Broadband Wireless Ad Hoc Networks , R. Ganesh, S. Kota, K. Pahlavan and R. Agust, Eds. Kluwer Academic Publishers, Chapter 9, pp. 141-161, 2004.