Wireless Pers Commun (2010) 52:423–447
DOI 10.1007/s11277-008-9636-5

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

Yosi Ben-Asher · Sharoni Feldman · Moran Feldman · Pini Gurfil

Published online: 4 December 2008

Abstract When studying scalability in ad-hoc networks, most works present experimental results for a limited number of nodes (100–200). Various “explicit” clustering techniques have been proposed to improve scalability, obtaining successful sessions for 400–800 nodes. However, explicit clustering may damage the performance, e.g., cause session breaks due to fast movements of cluster heads. An alternative to explicit clustering is the use of algorithms that are “naturally clustered”, i.e., arrange the nodes in dynamic hierarchical structures. In this work, we study the effect of explicit clustering by comparing an advanced version of the Ad Hoc Distance Vector Algorithm (AODV) with the Metrical Routing Algorithm (MRA) that possesses the natural clustering property. We cover fundamental aspects of scalability and experimentally prove the superiority of implicit clustering over explicit clustering. In particular, we consider heterogeneous theaters with several types of transmitters including personal, car-mounted, helicopters and a Geostationary (GEO) satellite. Natural clustering is more effective in heterogeneous theaters as the more powerful transmitters can serve as cluster heads. A formal bound based on general probabilistic assumptions shows that all existing ad-hoc algorithms cannot scale infinitely, thus rendering scalability as an experimental issue.

Keywords Ad-Hoc · Routing · Scalability · MRA · IFAS · Ad-Hoc simulator · Blocking factor

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

Y. Ben-Asher (✉) · S. Feldman · M. Feldman
Computer Science Department, Haifa University, Haifa, Israel
e-mail: yosi@cs.haifa.ac.il

S. Feldman
e-mail: sharoni@cs.haifa.ac.il

P. Gurfil
Faculty of Aerospace Engineering, Technion—Israel Institute of Technology, Haifa 32000, Israel
e-mail: pgurfil@technion.ac.il

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 in the number of successful sessions proportionally to the number of transmitters (nodes) in the theatre. For example, in [1] it is stated that "it has been proven that current routing protocols work well in small size networks (e.g. fewer than 100 nodes)". Reference [2] presents the Extended Hierarchical State Routing (HSR) and Extended HSR (EHSR) protocols with a network which consists of 100 nodes in a 1 kilometer (km) $\times$ 1 kilimeter theatre. Reference [3] 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 a few hundreds nodes and sessions. A formal argument (which we give later on) can be used to show that even for a very low number of session initiations, any existing ad-hoc algorithm will, for some sufficiently large number of nodes, become blocked and therefore very few sessions will reach their destinations. Hence, if the above bound is true, then scalability in ad-hoc network is a matter of experimental comparison between various algorithms. In particular, scalability involves comparing different types of deterioration curves of the number of successful sessions under various settings such as capacities, transmission ranges and velocities.

The use of explicit clustering is part of almost all existing ad-hoc algorithms for large scale networks. We show that explicit clustering techniques are not necessary if we consider another type of algorithm based on virtual coordinates rather than on dynamic learning of routing tables, as is the case with almost all proposed ad-hoc algorithms.

As indicated before, the common routing method of increasing scalability is the use of hierarchical/clustered protocols such as the Hierarchical State Routing (HSR) protocol [4] or the Intelligent Hierarchical State Routing protocol (IHSR) [5]. 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) than the remaining nodes in each cluster. Typically, cluster heads may be Helicopters, Unattended Aerial Vehicles (UAVs) or Cars with increased transmission range and capacity yielding a "heterogeneous theater" (see Fig.1). The use of selected cluster heads reduces significantly the traffic of packets in the network since communication between nodes of different clusters is restricted to the cluster heads backbone. Quoting [5], "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". We thus classify this type of algorithmic solutions as "explicit clustering" as cluster heads are explicitly selected.

Explicit clustering has several drawbacks: (1) there is a significant overhead in the maintenance of 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 adjacent 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 fragmentation of the theatre into clusters may result in a number of clusters which is larger than the number of the optimal one; (5) the session path may require more nodes than a direct path; (6) swapping of cluster heads results in routing changes and hence generates routing overhead. Cluster

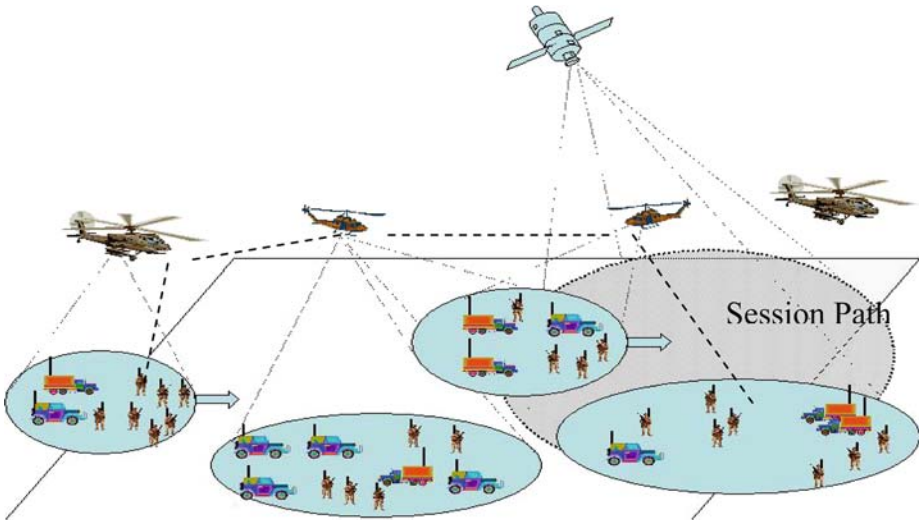


Fig. 1 Heterogeneous theater with an ad-hoc network layout

heads are exposed to rapid changes since the tendency is to elect the stronger nodes like UAVs to serve as cluster heads.

In this work, we study a different type of clustering named "implicit clustering" wherein cluster heads are not explicitly selected but rather "implicitly" emerge due to dynamic formation of dense areas. We thus consider routing techniques that can implicitly cause nodes that are in the "center" of dense areas to act as cluster heads. This ability is also referred to as "natural clustering" since there is no need to add explicit clustering mechanisms to such routing algorithms. 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 explicit clustering thus solving most of the above mentioned drawbacks of explicit clustering. The MRA is unique compared to most types of ad hoc networks algorithms since 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 advanced variant of the AODV protocol which is a core algorithm for ad hoc networks and is used by most works in the field for comparison purposes.

The main contributions of this work are:

- Studying the issue of implicit clustering versus explicit clustering.
- Finding an upper bound for the number of successful sessions in a given theatre.
- A detailed set of experiments covering a broad set of scalability issues in a heterogeneous theater of ad hoc network.
- Validating 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.

2 Background and Related Works

Clustering is extremely helpful for the AODV algorithm, which is a core algorithm for ad hoc networks. The AODV was proposed by Perkins and Royer [7]. 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 via dynamic updating of routing tables. If a route fails, the source node will attempt to establish an alternative path to proceed with a given session. The AODV creates the foundation for a variety of protocols in particular with different forms of explicit clustering techniques. For example, the Adaptive Routing using Clusters (ARC) [4] or Mobile Backbone Network Protocol-AODV (MBNP-AODV) [8] that utilizes the basic AODV with a clear distinction between backbone nodes (similar to cluster heads), regular nodes and nodes that can become backbone nodes.

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 heterogeneous transmitters introduces the need to scale the network without compromising performance. There are many references in the literature dealing with 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. Reference [9] defines the scalability as “the ability of a network to adjust or maintain its performance when the number of the nodes increases”. Reference [5] describes a heterogeneous network where unmanned aerial vehicles (UAVs) are used to bridge between ground mobile entities.

A common method is to split the 3D space into layers of different networks. The Extended Hierarchical State Routing (EHSR) [2] 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 regarding situations wherein there is a lack of backbone nodes in some regions of the theater or crowding of backbone nodes in other regions. The Intelligent HSR (IHSR) [5] 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 an heterogeneous network with three types of radio interfaces.

Scalability simulations are a crucial element for proving the ability of the proposed protocol to support large number of nodes. Many protocols are tested on small theaters with a small number of nodes. References [2,5] do not include any description of scalability simulations and the results of such experiments. Reference [10] 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. Reference [2] deals with a heterogeneous network that can scale up, but the simulations were performed with a fixed theatre of 100 nodes, a theatre size of 1 km $\times$ 1 km and a very slow and fixed speed of 2 m/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 [1]. The mobile nodes speed was selected to be in the range 0–10 m/s, with a large network of 1,000 nodes. The nodes were equipped with two radios—one is similar to every ordinary node with limited transmission range and the other with extended range for backbone communications. The simulations do not include scalability tests. Reference [11] reports on simulations performed to compare the Hierarchical landmark protocol (H-LANMAR) with flat LANMAR and flat AODV. The simulations include up to 36 backbone nodes with a single UAV connected to all backbone nodes in a theatre of 3.2 km $\times$ 3.2 km. No results were given on the scalability tests.

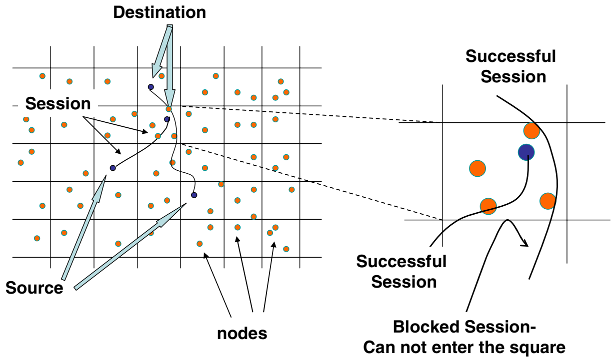


Fig. 2 Successful sessions and blocked session

3 The Upper Bound on the Number of Successful Sessions

Scalability in ad-hoc networks implies that for a field with n nodes in a given density and uniform distribution 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 $I(n)$, e.g., $S(n) = I(n)/8$. We show in the sequel that even for modest values of $I(n)$, this aim is impossible and in fact it is highly probable that almost all sessions will be blocked and will never reach their destinations. Scalability is possible only when $I(n)$ takes very small values relative to n.

To that end, let c be the capacity of every node, i.e., the number of sessions that can pass through any given node. Clearly, for $c = 1$, a single session with a long path will block all other sessions from crossing the field from one side 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 square area containing $\sqrt{I(n)} \cdot \sqrt{I(n)}$ nodes. Since we assume uniform distribution of the nodes in the field, there are roughly $\sqrt{I(n)}$ nodes on each one of the square's edges. This limits the number of successful sessions in this case to $4\sqrt{I(n)} \cdot c$, since this is the maximum number of sessions that can cross the boundary of this square. Thus, the proposed bound must be examined in a setting wherein there is a potential way to route all $I(n)$ sessions to their destinations; yet any "reasonable" algorithm will fail to find these routes. Finally, another reasonable requirement is that the bound will hold for $I(n) < c\sqrt{n}$ as there can be at most $c\sqrt{n}$ sessions of length $\sqrt{n}$ or more.

Figure 2 presents an example of the terrain divided into squares. In the enlarged square, we see two successful sessions, one is terminated inside the square and the second terminates in the square above the enlarged square. A third session coming from the square bellow the enlarge square is blocked as there are no free available resources.

The following assumptions are used by the proposed bound:

1. There are n nodes uniformly distributed in the plan such that it can be assumed that a sufficiently large area of size A contains A/λ nodes, where λ is the density of the field. We also assume that the transmission range is selected in such a way that for any sufficiently large square with m nodes there are $\sqrt{m}$ nodes which can communicate with nodes outside the square. All sessions must enter the square or exit it (or do both) through these nodes on the edges.
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 is exhausted the node is blocked and will ignore/reject any messages of sessions other than those for which the bandwidth was allocated.
4. Sessions are established by some process that sequentially allocates bandwidth on each node along a path from the source to the destination till it reaches the destination itself or fails in an intermediate node due to lack of bandwidth. When the process reaches a node without available bandwidth, the session will become blocked and some process of resources release will begin. We assume that the release process is initiated in the blocked end of the path and propagates backward towards the source of the session. The propagation takes a significant amount of time, and while it is taking place other sessions might be blocked by the part of the allocated path that has not been released yet.
5. We will divide (logically) the field into $k \cdot k$ squares (k > 3). We are interested only in sessions that must have a long path; therefore we assume that each session must pass through at least $k - 2$ intermediate squares between the squares of its source and destination nodes. The first assumption we make about the underlying algorithm is that if we select the size of a square to be small enough and that the source/destination of each session is random then the nodes through which a session path will enter and exit each one of the intermediate squares are random. This assumption is supported by the following arguments:

   - This assumption holds if we assume that each path is a random line in the 2D plan, as a random straight line passes through random places in every sub square we take.
   - Since the squares are small it is reasonable to approximate inner square paths as random paths.

6. We assume that the path allocated for a session within an intermediate square can be determined by the nodes by which the session enters and exits the square. This means that if two sessions s and s' enter and exit a square on the same end points, they will share the same path within this square. This is the only questionable assumption we make on the way sessions are routed to their destinations. Intuitively we assume that the underline algorithm routes 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 Fig. 3. A fully blocked square is presented in Fig. 4.

Let us look at a square containing d^2 nodes. We know that each session crossing this square must enter through one of the d nodes in one side of the square and leave through one of the d nodes in the other side of the square. This means that the session will have one of d^2 possible paths. The square will be blocked if c sessions will share the same path. Calculating the probability of this event is complex, therefore we will limit ourselves to the case $c = 2$. In this case the scenario becomes equivalent to the known "birthday paradox" where the d^2 paths represent the possible birthdays and the sessions represent the children. The probability of the square to be blocked with s sessions passing through is approximately $\frac{s^2}{d^2}$ for $s << d^2$. For the proposed bound we will use $s = \sqrt{\log(n)} \cdot \sqrt[8]{n}$ and $d = \sqrt[4]{n}$. Since s is much smaller than d (let alone d^2), the blocking probability of a square is $\log(n)/\sqrt[4]{n}$.

We partition the field to $k = \sqrt{n}$ squares each containing $\sqrt{n}$ nodes and having $\sqrt[4]{n}$ nodes on each of its edges. We will also set $I(n) = \log(n) \cdot \sqrt[8]{n}$ and $c = 2$. The sources and the

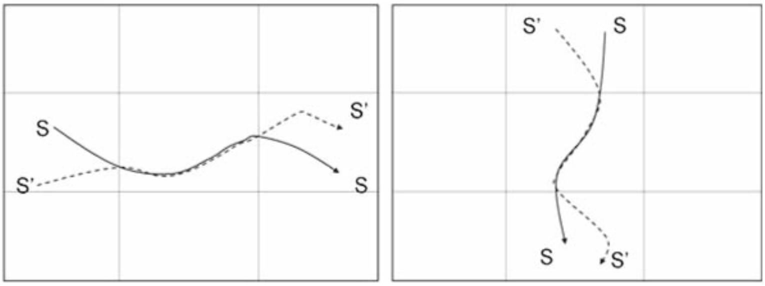
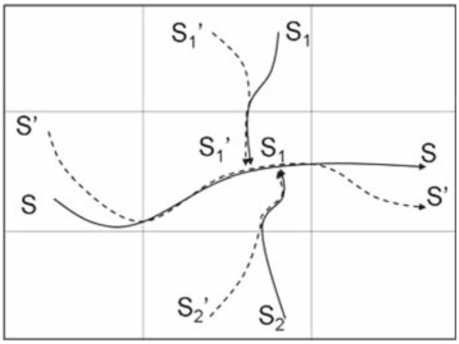


Fig. 3 Horizontal and vertical blocking

Fig. 4 Blocking square



destination are selected randomly so that each session must pass through $\sqrt[4]{n}$ squares, which is the number of squares a session must pass from one side of the field to the other. 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 cross is less than $\left(1 - \frac{\log(n)}{\sqrt[4]{n}}\right)^{\sqrt[4]{n}} \cong e^{-\log(n)} \cong \frac{1}{n}$. Thus with high probability almost all of the $I(n)$ sessions are blocked in one of the squares.

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 $\sqrt[4]{n} \cdot \sqrt[4]{n}/\lambda$ square as discussed earlier. With high probability, this destination square will be a blocking square, blocking most of the sessions. The fact that the bound holds for a 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 occurs for relatively large values of n, when the distance between the source and destination of every session exceeds some value r, $c = 2$ and $I(n) = \sqrt[4]{n}$. If we increase $I(n)$, under such conditions, there should be a decrease in $S(n)$. This phenomenon has not been observed before, because usually ad-hoc experiments use small values for n and other parameter values that make it more likely for $S(n)$ to degrade due to high values of $I(n)$ (in regard to n) and high moving speeds. The contribution of this bound is in showing that deterioration of $S(n)$ can occur even for a very low number of attempts to create sessions under some conditions. The following results presented in Fig. 5 show that this phenomenon occurs in actual simulations. The simulations were performed in a theater of 8.7 km × 8.7 km hosting 1,500 nodes with the restriction of minimal path length = 5 km. The left side of Fig. 5 presents the results for $c = 2$ and the right side for $c = 3$.

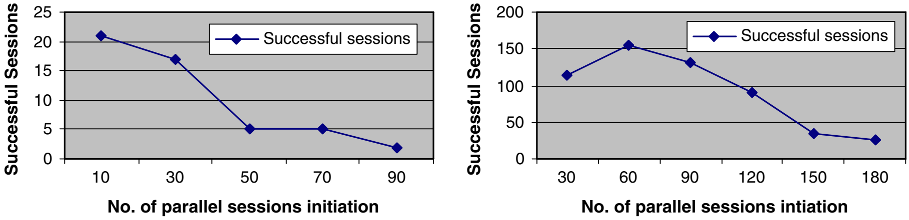


Fig. 5 Performance deterioration for $c = 2$ (*left side*) and $c = 3$ (*right side*)

The proof presented above assumes that $c = 2$. It can be easily seen that although increasing this value increases connectivity, the situation is not significantly different. Another assumption is that the transmission radius r has such a value that makes each square of m nodes "see" $\sqrt{m}$ nodes of other squares. If we increase r to have a value so that the previous square will "see" $\alpha\sqrt{m}$ for some constant $\alpha > 1$, then by multiplying $I(n)$ by α as well we will preserve the ratio $\frac{s^2}{d^2}$, which is the blocking probability of a square, and get the same result with a slightly larger $I(n)$. The final conclusion of this bound is that for all existing algorithms and all possible settings a blocking of the number of successful sessions will occur for sufficiently large values of $I(n)$. Under these conditions, scalability is measured experimentally by comparing "deterioration curves" of the form described in Fig. 5.

4 The Metrical Routing Algorithm

The Metrical Routing Algorithm (MRA) protocol [6] is classified as a virtual coordinate protocol. The algorithm approximates distances on the Euclidian plain by embedding the node's connectivity/radius-graph[1] in a tree like metric as depicted in Fig. 6a. Note that the tree only approximates distances. For example, the distance of nodes e and b is 1 in the 2D plain but is distorted to 3 in the tree metric (Fig. 6b). This approximation pays off since distributed update of coordinates is possible and involves few corrections of coordinates. For example if node g in Fig. 6b moves closer to (into the transmission range) nodes e or d then we only have to update g's coordinates determining its father in the big tree (d or e). In general at any stage each node belongs to one tree representing an approximate metric of the group of nodes connected by this tree. A unique property of the MRA algorithm is that the actual distance between two nodes may be smaller than the path length in the tree since the MRA algorithm may use original edges of the radius graph not included in the tree's edges. For example routing a session from b (with coordinates 0.1) to e (0.3.1) will not go through the root and use the original edge of the radius graph as a "shortcut". Note however, that the metric defined by using spanning trees + shortcuts is only an approximation to the actual distances of the radius graph. This is because a shortcut edge $u \rightarrow v$ is used for a session destined to node t only when the distance on the tree (path length) from v to t is shorter than the tree distance from u to t (see Fig. 6c). Thus the tree + shortcuts metric is a novel metric (not considered before) especially suitable for the dynamic setting of ad hoc networks. Note that the only information we need to implement the metric per node is its coordinates, a

[1] Let G be a graph in which each node has an associated location in a metrical space. G is called radius graph if there is a constant r such that there is an edge between two nodes u and v if and only if the distance between u and v does not exceed r. Looking at a group of nodes in an Euclidian plain it can easily seen that the graph formed by connecting each two nodes by an edge if and only if they are within transmission range is a radius graph.

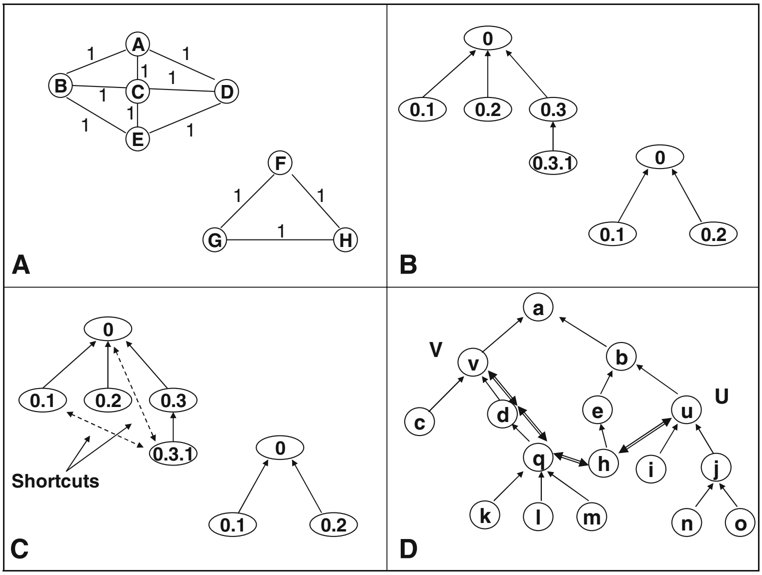


Fig. 6 Tree metric & shortcuts

pointer to its father in a tree and the unique id of the tree it is currently connected to.[2] The use of shortcuts eliminates possible bottlenecks at the tree root.

Dynamic changes in the radius graph resulting from rapid movements of nodes in the plain require constant distributed updates of the current coverage by the tree + shortcuts metric. The goal is to cover the plain by the minimal possible number of trees (i.e., number of connected components of the current radius graph). Let u be a node which belongs to tree $T1$ and v a separate node belonging to a different tree $T2$ and assume that node v moves into the transmission range of node u. The goal now is to efficiently find a joined tree + shortcuts metric for the two groups of nodes in $T1$ and $T2$. Obviously this can be done in several ways hence we seek the most simple and efficient way to do so. Note that at the same time other nodes in $T1$ and $T2$ may initiate a metric-join process with other nodes of different trees. Even for two trees it is hard to find an efficient fully distributed algorithm that will find the optimal metric for $T1 \cup T2$. Thus for single nodes we define the metric-join processes as follows:

1. Assuming that $T1$ has more nodes than $T2$, it makes sense to "move" v from $T2$ to $T1$ by becoming the next son of u updating its coordinates accordingly.
2. The joining node v may still be connected to its father and/or some of its sons in $T2$ thus v will send them a message instructing them to "join" $T1$ through itself. Note that if other nodes in $T2$ are engaged in a similar process then $T2$ may be separated to several components each joining a different tree. An example of "one frame" in such a case is presented in Fig. 7.
3. If v has several new neighbors it will select the one with the maximal estimated number of nodes to join with.
4. Separate processes accumulate estimation to the number of nodes in every sub tree.

[2] We also maintain an estimation of the current number of nodes in the tree.

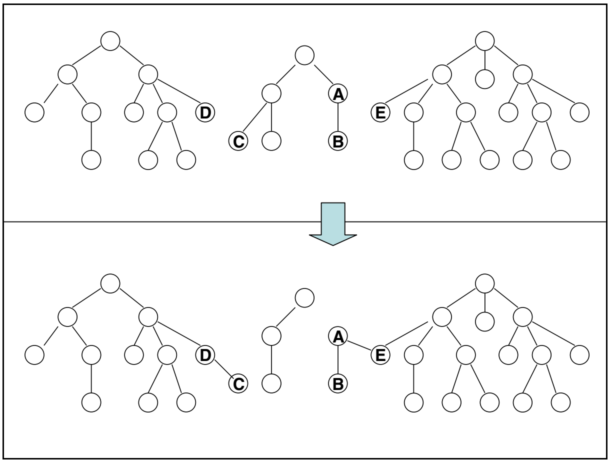


Fig. 7 Split & join of trees

This process is depicted in Fig. 6d. The metric-join processes do not affect in any way or break any active sessions. The MRA includes a distributed implementation of a directory to find the coordinates of every node so that routing of sessions can be performed using the shortest path in the current tree + shortcuts metric.

Every session occupies a source, a target and if necessary, transit nodes. The source is the proactive side initiating the session. The target and the transit nodes react to the session initiation request. The source is in charge of resuming the session after a session break. The *path allocation* procedure starts immediately after the source gets the target address. This process creates the session path from the source to the target. This process can fail because of various reasons, such as a target node which is busy with a terminating session or a transit node with unavailable resources for a transit session. A *packets transfer* mechanism is utilized as soon as the path allocation process succeeds 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 packets for a while, it assumes that the path has been disconnected and it clears the call from its tables. When the source notices that the path is disconnected, it initiates a new path-finding process.

5 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.

5.1 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.

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 8 presents the entities management screen. It enables the user to define any number of entities in the field and control their behavior.

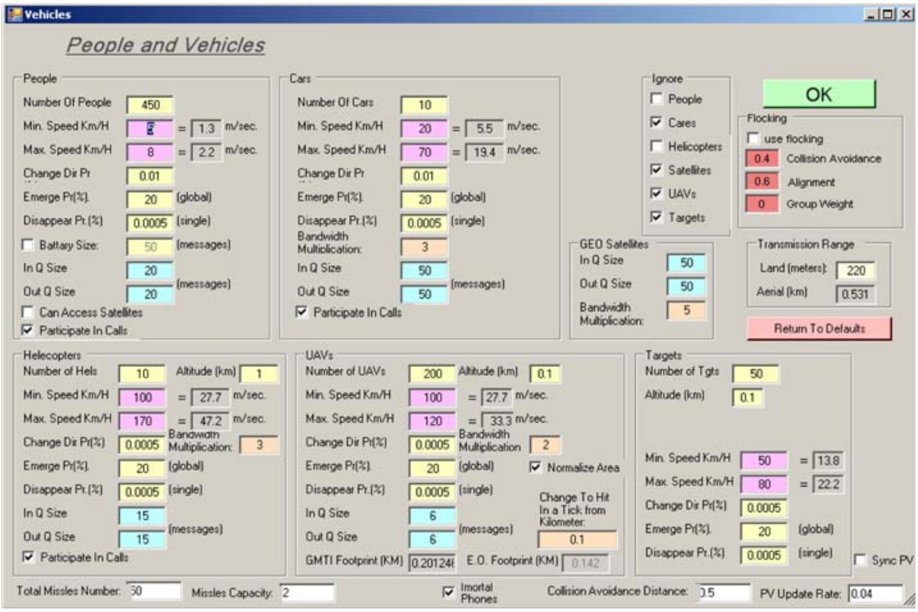


Fig. 8 Entities (people, cars, helicopters and UAVs) management screen

5.2 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 that can reach the size of 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 3.5 km × 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 s. The maximum session setup time is limited to 0.7 s. The simulator tries to recover from an unsuccessful session setup or a session break. The number of retries can be adjusted.

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 s. As opposed to other nodes, the satellite cannot generate or terminate sessions.

Table 1 depicts global parameters common to *most* simulations.

Table 1 Simulator parameter values

Attributes	Selected Values
Node Bandwidth	200 Kilo-bits (kb)/second (s) A single session bandwidth is 20 kb/s
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/s	33 messages per session/s
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	Adjustable: satellite: in/out-queue: 50 messages; persons, cars, helicopters: in-queue: 20 messages; out-queue: 20 messages
Medium access control (Mac) model	802.11 standard [12] was implemented, mainly handling collisions of overlapping transmissions
RF propagation	$Loss = 92.5 + 20 \times \log(d \times f)$ In the free space. Near ground—$loss = 40 \times \log(d) - 20 \times \log(Ht \times Hr)$ where d is distance between antenna in meters, f is GHz , Ht/Hr is height of transmitter/receiver in meters.

5.2.1 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.

5.2.2 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).

6 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:[3]

- The MRA algorithm is naturally clustered so that hierarchical clustering is created without using explicit clustering techniques. Unlike the AODV algorithm it does not require 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 interested in comparing deterioration curves 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.

6.1 Experiment Set 1 — Scalability in the Number of Nodes

The scalability (i.e., deterioration curves) of the MRA compared 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 sessions is measured. The results show that for the MRA the deterioration curve is significantly better than those of the AODV, hence MRA scales better. These results repeat themselves for any combination of persons, cars and helicopters, in fact up to certain values increased heterogeneity for the MRA seems to increase its performance.

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

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

Table 2 Persons successful sessions: MRA Versus AODV ($c = 10$)

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	–	–

exists). Finally, for a given number of persons there is 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) too much. This is because adding too many helicopters and cars increase queues' overflow and packet loss.

Table 2 presents in the left side of every cell the number of successful sessions for increasing numbers of personal transmitters (100–460) 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 then a certain percentage of the packets passed. The results show that until the deterioration stage, 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 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 passes a certain threshold. In this experiment the MRA's performances starts to drop when the density becomes greater than 40 transmitters per square km.

It is interesting to note that practically for some limited range of densities, capacity and transmission range we can avoid the deterioration curve. Figure 9 Presents the results of scalability tests of the MRA with a network growing from 100 to 2,000 nodes with a fixed density (i.e., adjusting the field size proportionally to the number of nodes). The deterioration and blocking have been prevented due to: (a) use of large capacity ($c = 10$), (b) allowing many sessions to be relatively short, and (c) using a large transmission radius so that each node has about eight neighbors instead of four. The results present a constant and linear 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.

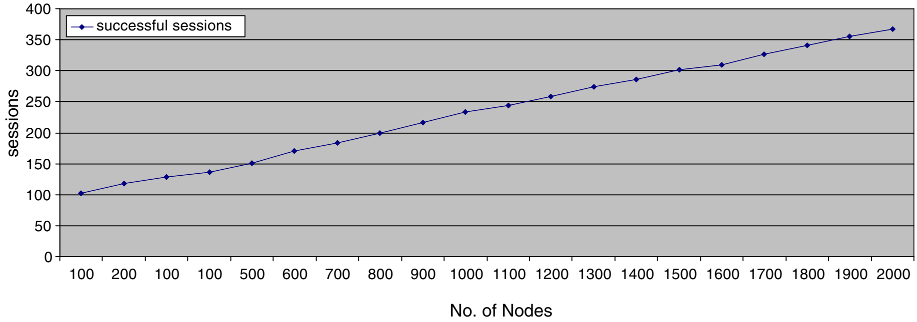


Fig. 9 Scalability of the MRA with fixed density

6.2 Experiment Set 2—Explicit Clustering

The implicit 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

$$person \Longleftrightarrow person \Longleftrightarrow car \Longleftrightarrow helicopter \Longleftrightarrow helicopter \Longleftrightarrow car \Longleftrightarrow person \Longleftrightarrow person$$

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

$$person \Longleftrightarrow helicopter \Longleftrightarrow person \Longleftrightarrow person \Longleftrightarrow car \Longleftrightarrow person$$

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

The heterogenic theater creates a natural hierarchy. Figure 10 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:

1. Figure 10a 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% 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.
2. Figure 10b presents the case in which helicopters are replaced by cars whose transmission range is the same as personal transmitters but can handle tripled capacity. The analysis of Fig. 10b 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 car

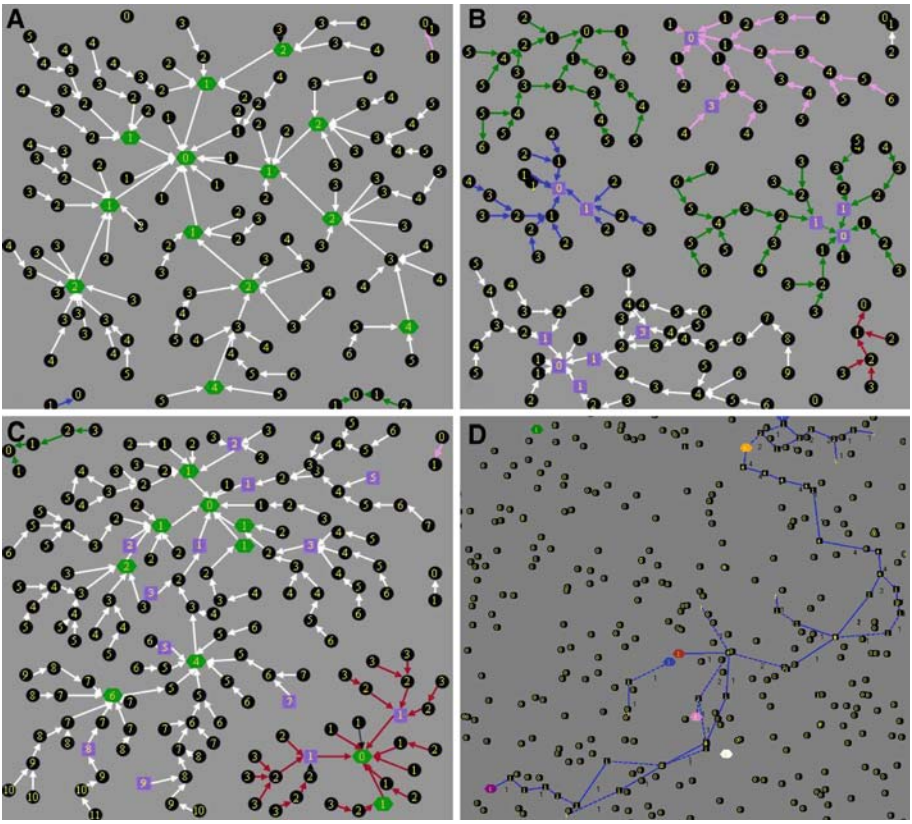


Fig. 10 MRA trees and AODV screenshots

and helicopter bandwidth is much broader. As a result, fewer packets are lost. Some 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.

3. Figure 10c 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.
4. Figure 10d 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 implicit 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.

Table 3 Percentage of clustered sessions from X hops sessions

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%

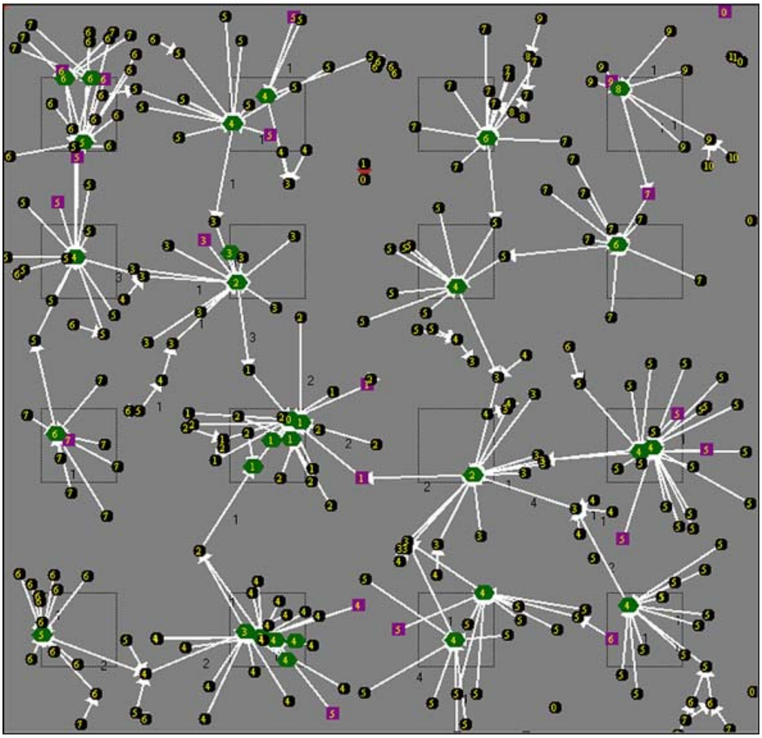


Fig. 11 Artificial clusters (ACLs) in theater

6.3 Experiment Set 3—Artificial Clustering Versus Explicit Clustering

Another set of experiments demonstrating explicit clustering versus artificial clustering 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 11 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.

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

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

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

Table 5 MRA and AODV average number of legs

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

Table 6 Successful sessions in heterogeneous theatre with persons and cars

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

Table 7 Successful sessions in heterogeneous theatre with persons and Helicopters

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

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.

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

Tables 6 and 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.

6.4 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 two 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 increase of the transmission range and bandwidth improves the number of successful sessions for both AODV and MRA (so both algorithms scale up), however the

Table 8 Medium density—no. of successful sessions— transmission range versus link capacity

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 9 High Density—no. of successful sessions— transmission range versus 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

MRA scales approximately twice better than the AODV. Again this advantage of the MRA is explained by the implicit 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” 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 implicit 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 (Tables 8, 9).

6.5 Experiment Set 5—Speediness Versus 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 implicit clustering (ICL) 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) The average

Table 10 Entities speeds

Entities	No	Movement speed (km/h))		
		Slow	Medium	Fast
Personal transmitters	300	1–2	4–5	8–10
Cars	10	20–30	45–55	70–120
Helicopters	10	20–30	90–110	170–230

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 ICL 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 implicit clustering and also the dependency of the AODV in clustering techniques to overcome high densities. Note that in average for both ACL and ICL 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.

6.6 Experiment Set 6—Reverse Effect

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 clustered does not enforce such a restriction. Consequently, as demonstrated before (Fig. 11) the MRA functions better with ICL 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.

Table 11 Variable speed results

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
		ICL	161	138	125	104	5.8
	Medium movement	ACL	101	91	78	65	6.3
		ICL	138	107	98	83	5.9
	Fast movements	ACL	40	40	29	18	5.5
		ICL	63	62	60	48	4.6
AODV	Slow movement	ACL	84	78	65	44	6.7
		ICL	74	73	34	19	6.2
	Medium movement	ACL	49	46	34	33	6.6
		ICL	43	38	31	17	6.2
	Fast movements	ACL	15	15	11	7	3.8
		ICL	12	11	9	3	3.8

Table 12 Reversed hierarchy results

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

6.7 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 Fig. 12 the GEO satellite covers most of the theatre and creates a global cluster head. A helicopter can communicate directly with 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.

We tested four cases presented in Fig. 13. 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 to adding one or two helicopters.

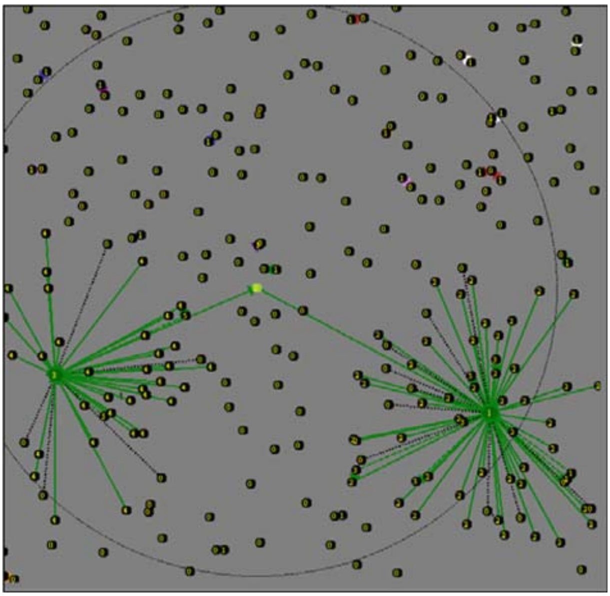


Fig. 12 A theater covered by a satellite (in the center)

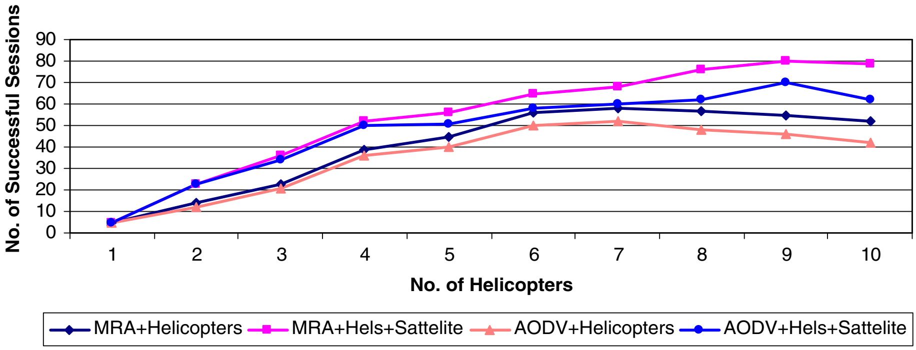


Fig. 13 Substitution between helicopters and satellite

The contribution of the satellite is therefore not in its ability to increase the number of successful sessions but rather (as depicted in Fig. 13) 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 improves when the satellite is added. The decrease in the number of successful session when the number of helicopters grows over eight, results from the queues overflow and the inability of the MAC layer to transfer successfully all messages.

7 Conclusions

We described in this research a study of complex nodes and group behavior in an ad-hoc network constructed of heterogeneous 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.

We consider the use of explicit clustering mechanisms versus implicit clustering (or natural clustering). The use of explicit clustering wherein cluster heads are explicitly selected is part of almost all existing ad hoc algorithms. We show that explicit clustering techniques are not necessary if we consider an algorithm like MRA which is based on virtual coordinates with natural clustering capabilities.

A formal lower bound argument is given showing that any ad-hoc algorithm that does not use load adaptive routing can not scale. Load adaptive techniques are probably not suitable for ad hoc networks as load information and location of hot-spots change rapidly. Consequently scalability of ad hoc algorithms is mainly subject to an experimental proof measuring the point where the algorithm ceases to scale. We showed that the MRA maintained a scalability curve for larger number of nodes compared to the AODV (enhanced by explicit clustering). The AODV is used as a representative for many other types of ad hoc algorithms. The MRA scaled well for 2,000 nodes while the enhanced AODV ceased to scale for a few hundred nodes. We believe that this occurred due to the implicit clustering of the MRA. The experiments covered a wide range of aspects including:

1. Pure scalability measurements, i.e., number of successful sessions versus increasing number of nodes (same type).
2. Heterogeneous scalability, namely how well the underlying two algorithms scale up when using growing 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.
3. Proving explicit clustering abilities of the MRA by showing that the MRA naturally uses the more powerful nodes as cluster heads.
4. Measuring the “reverse effect” of explicit clustering wherein some sessions bypass the cluster heads.
5. Measuring the benefit of using artificial clustering for both the MRA and AODV. It turns out that as expected clustering is essential for the AODV’s scalability but only limits the scalability of the MRA.
6. Relative effects of increased transmission range versus increased capacity.
7. Measuring the effect of the nodes speed on the scalability.

References

1. Xu, K., & Gerla, M. (2002). A heterogeneous routing protocol based on a new stable clustering scheme. In *Proceedings of MILCOM 2002* (Vol. 2, pp. 838–843). Available from www.cs.ucla.edu/NRL/wireless/uploads/kxu_milcom02.pdf.
2. Lihui Gu, D., Pei, G., Ly, H., Gerla, M., & Hong, X. (2000). Hierarchical routing for multi-layer ad hoc wireless networks with UAVs. In *Proceedings of Milcom*.
3. Li, J., Hass, Z. J., & Sheng, M. (2002). Capacity evaluation of multi-channel multi-hop ad hoc networks. In *TCPWC*.
4. Blending-Royer, E. M. (2002). Hierarchical routing in ad hoc mobile networks. *Wireless Communication & Mobile Computing, 2*(5), 515–532.
5. Gu, D. L., et al. (2000). UAV-aided intelligent routing for ad-hoc wireless networks in single-area theater. In *IEEE WCNC*. Available from http://citeseer.ist.psu.edu/gu00uav.html.
6. Ben-Asher, Y., Feldman, S., & Feldman, M. (2006). Ad-hoc routing using virtual coordinates based on rooted trees. In *IEEE SUTC*, Taiwan.
7. Perkins, C. E., & Royer, E. M. (1999). Ad hoc on demand distance vector routing. In *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications* (pp. 90–100), New Orleans, LA, February 1999.
8. Rubin, I., Behzad, A., Ju, H., Zhang, R., Huang, X., Liu, Y., & Khalaf, R. (2004). Ad hoc wireless networks using mobile backbones, Chap. 9. In R. Ganesh, S. Kota, K. Pahlavan, & R. Agust (Eds.), *Emerging location aware broadband wireless ad hoc networks* (pp. 141–161). Dordrecht: Kluwer Academic Publishers.
9. Villasenor-Gonzalez, L., & Ge, Y. (2005). HOLSR: A hierarchical proactive routing mechanism for mobile ad hoc networks. *IEEE Communication Magazine, 43*, 118–125.
10. Rhee, K. H., Park, Y. H., & Tsudik, G. (2004). An architecture for key management in hierarchal mobile ad hoc networks. *Journal of Communications and Networks, 6*(2), 156–162.
11. Xu, K., Hong, X., Gerla, M., Ly, H., Gu, D. L. (2001). Landmark routing in large wireless battlefield networks using UAVs. In *Military communications conference 2001*. Communications for network-centric operations: Creating the information force. IEEE (Vol. 1, pp. 230–234). Available from www.cs.ucla.edu/NRL/wireless/uploads/kxu_milcom01.pdf.
12. IEEE Computer Society. IEEE 802.11 standard, IEEE standard for information technology.

Author Biographies

Yosi Ben-Asher received his Ph.D. in computer science from Hebrew University in 1991. He is currently a lecturer at the Department of Computer Science, University of Haifa, Israel. His research interests include parallel systems, compilers, distributed WEB applications and reconfigurable networks. Current projects include: Highlevel synthesis, program merging, source level compilation, and distributed Ecommerce.

Sharoni Feldman received his B.Sc. degree in computer science from the Technion—Israel Institute of Technology and M.Sc. degree in Business Administration in the field of Information Systems from Tel Aviv University and Ph.D. from the Computer Science Department, Haifa University, Israel. From 2006 to 2008, he was with the Technion, Israel Institute of Technology in Aerospace Engineering from the Technion where he served as Research Staff Member. Dr. Feldman is a consultant to the Israeli Aerospace Industries (IAI) and other high-tech companies.

Moran Feldman recieved his B.A. degree in Computer Science from the Israeli Open University in 2004. He received his M.Sc. in 2008, from the Israeli Open University; in approximating minimum cost directed Steiner forests (Magna Cum Laude). Currently he is a Ph.D. student in the Technion Israel Institute of Technology in the Computer Science department.

Pini Gurfil received his Ph.D. in Aerospace Engineering from the Technion, Israel Institute of Technology, in 2000. From 2000 to 2003, he was with Princeton University's Department of Mechanical and Aerospace Engineering where he served as Research Staff Member and Lecturer. In September 2003, Dr. Gurfil joined the Faculty of Aerospace Engineering at the Technion. Dr. Gurfil is founder and head of the Distributed Space Systems Laboratory at the Technion. Dr. Gurfil has been conducting research in Astrodynamics, Distributed Space Systems, Vision-Based Navigation, Optimization and Multiagent Systems. He published over 120 journal and conference articles in these areas.