A Taxonomy of Location Management in Mobile Ad Hoc Networks

Laura Galluccio and Sergio Palazzo

Abstract: **Location management is difficult in ad hoc networks due to many features such as the lack of a wired infrastructure, the scarce energy, memory and processing capabilities of nodes, and nodes' movement which leads to a dynamic topology. These characteristics make the location management schemes designed for mobile cellular networks inefficient for ad hoc networks. New solutions for location management have therefore been proposed in the literature in the recent past. In this paper, a taxonomy of location management strategies is presented; some of the more interesting approaches proposed in the literature are critically discussed, highlighting their advantages and disadvantages.**

Index Terms: **Ad hoc networks, energy efficiency, location management, robustness, scalability.**

I. INTRODUCTION

Location management consists in the set of procedures responsible for the dissemination, storage, and retrieval of information about the position of nodes. This information is required by routing protocols to localize the destination of a data flow before performing a selective route search. Consequently, location management and routing represent two complementary functionalities which cooperate to support communications between nodes.

Location management is particularly difficult in ad hoc networks because

- Management procedures cannot rely on a wired infrastructure.
- Nodes have scarce capabilities in terms of energy, memory, and computational resources.
- Nodes in ad hoc networks are, in general, mobile and prone to failures, so the network topology continuously changes.

It is therefore obvious that the location management schemes used in traditional cellular networks cannot be successfully applied in ad hoc networks and appropriate new schemes are therefore needed. Desirable characteristics for such schemes are

- *Energy efficiency*: A limit to the amount of energy required to perform location management procedures is important since it allows network lifetime to be increased.
- *Low processing and storage*: Mobile terminals are likely to have limited processing and storage capabilities. The amount of location information to be processed and stored in each node should therefore be as limited as possible.
- *Robustness to node failures*: Location management schemes should continue to work in spite of possible node failures.
- *Scalability*: It is important to limit the number of location updates, as well as the size of location information tables stored by nodes, in spite of possible increases in the number of nodes in the ad hoc network.
- *Load balancing over all nodes*: The processing, forwarding, and storage of location information must be shared fairly between all nodes. If this is not the case, early battery consumption by overloaded nodes may result in a shorter network lifetime.

A number of surveys of location management solutions for ad hoc networks have appeared recently [1], [2]. In this paper, instead, we provide a taxonomy of location management schemes, trying to classify and critically discuss the most interesting and significant approaches provided so far in the literature. More specifically, in Section II a simple classification is introduced which distinguishes between *structured* and *unstructured* location management schemes. The first category is discussed in detail in Section III, whereas *unstructured* schemes are considered in Section IV. Finally, some concluding remarks are made in Section V.

II. TAXONOMY OF LOCATION MANAGEMENT SCHEMES

In mobile ad hoc networks, two classes of solutions to the location management problem can be distinguished

1. **Structured schemes**, where location information is forwarded to certain nodes selected on the basis of their identities or certain topological or structural characteristics, e.g., position, computational capabilities, and energy capabilities, etc. An example of this is a scenario where nodes are organized in groups and a given node is appointed as group representative and is responsible for management of the location information for all nodes in the group.
2. **Unstructured schemes**, where location information is forwarded to nodes independently of their specific identities and characteristics. A simple example is the case where location information is flooded throughout the whole ad hoc network. In this case, location updates are forwarded to all nodes.

Within the *structured* class, two categories of schemes can be distinguished between, depending on the procedure used to select the nodes to which the information should be forwarded. More specifically,

- *Non-adaptive* schemes select location databases according to an initial setting which remains unchanged, i.e., the set of nodes to which the location updates are sent is selected a priori, during the network initialization phase.
- *Adaptive* schemes distribute location updates to a set of nodes which can be dynamically changed. Within this cat-

---

Manuscript received July 31, 2004.

L. Galluccio and S. Palazzo are with the Dipartimento di Ingegneria informatica e delle Telecomunicazioni, University of Catania, Italy, email: {lgalluccio, spalazzo}@diit.unict.it.

egory an additional classification can be made, according to the technique used to select location databases which depends on

  - A node's current position. We call these approaches *position-based* schemes.
  - A node's reciprocal distance from other nodes. We call these approaches *distance-based* schemes.

Unstructured schemes can be further divided into two types of approaches depending on the attitude of nodes to disseminate and propagate location information. Accordingly, we can distinguish between

- *Introvert schemes*, where nodes do not disseminate or propagate location information spontaneously.
- *Extrovert schemes*, where nodes are very active in disseminating information about their location and propagating other nodes' location information.

Unstructured schemes can help to guarantee a fair distribution of the workload between nodes since all nodes are equally required to take part in the location update process. This can result useful in sensor network scenarios where the limited processing and storage capabilities of sensor devices require simple and light operations to be performed by all nodes. However, in many application scenarios, such as pure ad hoc networks or hybrid ad hoc/sensor networks, nodes exhibit different features in terms of available energy and computational resources. In this case, *structured* schemes could be more appropriate both because they may allow to mostly exploit the capabilities of more robust devices and so the terminal battery to last longer, and because they may achieve higher performance in terms of efficiency and reliability of the location management procedures.

III. STRUCTURED SCHEMES

In this section, we describe the basic characteristics of *structured* schemes. More specifically, in Section III-A we will focus on *non-adaptive* schemes while in Section III-B we will deal with *adaptive* schemes. We will highlight the advantages and disadvantages of each of the above categories and the scenarios where each approach is the most convenient.

A. Non-Adaptive Schemes

In these schemes, each node sends its location updates to a set of nodes defined a priori. Let $\Phi_{n'}$ be the set of nodes responsible for the location management of node n'. If node n'' needs information about the current location of n', it will query one or several nodes in $\Phi_{n'}$. Obviously, these schemes require node n'' to be able to determine at least one of the nodes belonging to $\Phi_{n'}$ and its position. To this end, two approaches can be distinguished

- First approach: The set of responsible nodes is independent of the node being searched for, i.e., $\forall n'$ and n'', the following relationship holds: $\Phi_{n'} = \Phi_{n''} = \Phi$.
- Second approach: The set of responsible nodes depends on the node being searched for, i.e., $\exists\ n'$ and n'' such that $\Phi_{n'} \neq \Phi_{n''}$.

In the first approach, all nodes send their location updates to nodes in Φ which constitute a sort of backbone. To this end, all nodes must be informed about the current position of the nodes in Φ. This can be achieved by periodically flooding the network with location information for the nodes in Φ. A node n'' which needs location information about node n' will therefore contact the nearest node belonging to Φ.

The major problem of this approach is the lack of scalability. In fact, the nodes in Φ have to manage location information about all nodes in the ad hoc network, which might result in an overload. This will be proportional to the number of nodes being managed and the mobility degree of nodes.

In the second approach, to solve this problem, different sets of responsible nodes can be defined for different nodes. In this way the workload is distributed more evenly. However, it is necessary, for any pair of nodes n' and n'', that node n'' is able to identify and locate at least one node belonging to $\Phi_{n''}$. In this context, two interesting solutions are proposed in [3] and [4]. In [3], location databases are selected using a uniform technique, while in [4] the selection is random. More specifically, location databases are divided into sets called *quorums*. A quorum is a set of databases containing replicas of a mobile node's location information. Let Γ_i denote the i-th quorum; according to the notation used so far, if location information for node n is stored in Γ_i, it follows that the set of nodes responsible for the location management of node n coincides with the set of nodes which constitute Γ_i, i.e., $\Phi_n \equiv \Gamma_i$.

Moreover, as shown in Fig. 1, quorums must be defined in such a way that the intersection between any pair of quorums is not empty, i.e.,

$$\forall i \text{ and } j, \qquad \Gamma_i \cap \Gamma_j \neq \emptyset. \tag{1}$$

A node belonging to the intersection between two different quorums stores location information for all the nodes registered in both quorums. As a consequence, in each quorum it is possible to find location information about all the nodes in the network. Therefore, if a node n' whose location information is stored in quorum Γ_i needs location information about node n'' registered in quorum Γ_j, it queries the nodes in its quorum, Γ_i, and two cases are possible

- *Case 1*: $i = j$, i.e., the two nodes belong to the same quorum. If this is the case, nodes in Γ_i possess the location information for node n''.
- *Case 2*: $i \neq j$, i.e., the two nodes belong to different quorums. If this is the case, there will be a location database, $\bar{n}$, belonging to the intersection between the two quorums, i.e., $\bar{n} \in \Gamma_i \cap \Gamma_j \neq \emptyset$, which will answer the inquiry.

As an example, in Fig. 1 three quorums can be identified

$\Gamma_1 = \{1,2,3,4,10,11\}, \Gamma_2 = \{7,8,9,10,11\}$, and $\Gamma_3 = \{3,4,5,6,7\}$.

Accordingly it can be derived

$\Gamma_1 \cap \Gamma_2 = \{10,11\}, \Gamma_1 \cap \Gamma_3 = \{3,4\}$, and $\Gamma_2 \cap \Gamma_3 = \{7\}$.

If node N1 needs location information for node N3, then an inquiry will be sent to all the nodes belonging to Γ_2. Node 7 belongs to both Γ_2 and Γ_3, which is the quorum of node N3; therefore, node 7 stores information about the current location of node N3 and consequently answers the inquiry issued by node N1.

This second approach is more scalable than the first one; however, two major problems still survive

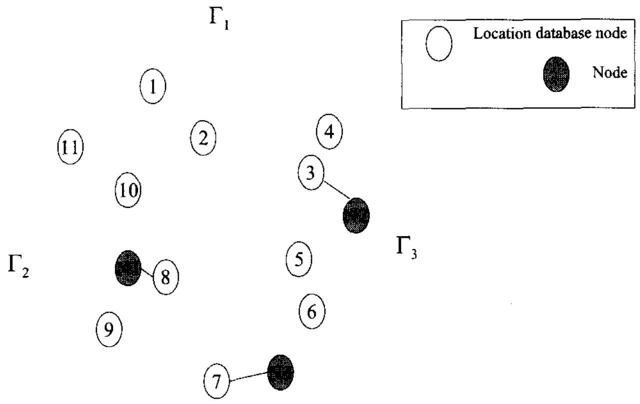

Fig. 1. Example of quorum structure.

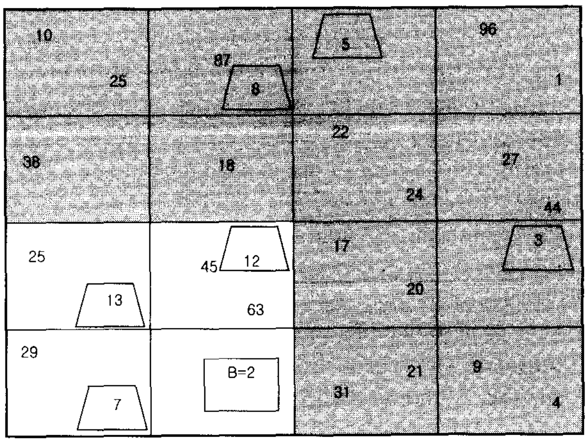

Fig. 2. GRID: An example of a 4-th-order square.

- Definition of the quorums is complex as the number of nodes increases.
- The robustness of this approach is vulnerable to failures and movements in location database nodes belonging to the intersections between quorums.

To cope with this problem, in [5] another quorum approach which takes dynamic network topology into account is proposed.

This second category of non-adaptive schemes results more complex than the first one. In particular, the overhead associated to the definition of quorums can be very large, depending on the size of the network and on the number of quorums being identified.

B. Adaptive Schemes

In this section, two categories of adaptive location management schemes will be discussed, highlighting their advantages and disadvantages. More specifically, position-based schemes will be described in Section III-B.1 while distance-based approaches will be presented in Section III-B.2.

B.1 Position-Based Schemes

In these schemes, the location information for a generic node n is stored by some location database nodes located in a given region that depends on n's position. We denote this region as A_n. Obviously, the set of location database nodes changes according to the position of n. Two approaches can be defined for identification of the region A_n

- First approach: The region A_n depends on the identity of node n itself, see for example [6]–[8].
- Second approach: The region A_n depends on the current position of n, see for example [9].

As an example of the first approach, in [6], A_n can be thought of as the *home region* of n. On the one hand, in fact, node n sends its location updates to the nodes located in A_n and on the other hand a node n' which needs information about the current position of n will query the nodes in A_n based only on information about node n's identity, which can be an indicator of its initial position. To this end, it is necessary for n' to be able to derive A_n correctly. In [6] this is achieved by using a *hash function* which maps a point in the space to the ID of node n. Obviously, the delivery of location update messages, as well as inquiry messages, requires appropriate geocast routing protocols [10]. The major advantage of this approach is that, if the hash function is well designed, high scalability can be guaranteed. However, the main drawback, due to node mobility, is that there may be no nodes within the home region of a certain node n. If this is the case, the location updates and location inquiry messages may be lost and recovery of the location information for node n must be achieved by means of more robust schemes, such as *unstructured extrovert* schemes (see Section IV-B).

Moreover, if a node is far from its home region, location update messages have to travel through a large number of intermediate nodes, which results in a large signaling overhead and long update delays.

Other works which follow the same approach have been proposed in [7] and [8]. In these cases, a set of nodes can be selected which store location information for each node n having its own identifier. We call such nodes *location servers*. Note that different nodes generally have different location servers. More specifically, in [7] and [8] the network area is partitioned into a hierarchy of circles or grids, respectively. For the sake of simplicity, in the following description of the mechanism used we will refer to [8]. In this case, as shown in Fig. 2, squares of increasing size are considered. Each square of order k contains exactly four squares of order $(k-1)$. The smallest squares are called the 1st-order squares. The location servers for a certain node B are selected in GRID [8] as follows. Let K be the highest order of squares defined in the grid. For any layer of hierarchy k, with $1 < k \leq K$, there are only 4 squares of hierarchy $(k-1)$ contained in the square of order k where B is located. Node B will have one location server in each of the three squares of order $(k-1)$ that do not contain B. In each of the above squares the location server will be the node whose identifier, ID_{LS}, is the minimum such that $ID_{LS} > ID_B$, where ID_B is the identifier of node B. Observe that, according to this scheme, three location servers are required for each hierarchy of squares. For example, referring to Fig. 2 and considering a square of order $K = 4$, there are 4 squares of 3rd-order, each one composed of 4 squares of 2nd-order. Accordingly, the servers which will retain information about B are those indicated, i.e., 7, 12, and 13 in the order-2 squares and 3, 5, and 8 in the order-3 squares.

In this approach, as soon as a node's location changes, its location servers must obviously be updated. However, only 3 nodes per hierarchy are required to store information about the node's position so that this solution appears to be very scalable.

Concerning the second approach, in [9] the region covered by the ad hoc network is divided into several areas, $A^{(1)}, A^{(2)}, \cdots, A^{(K)}$. For any area $A^{(k)}$, with $k \leq K$, one node, $n^{(k)}$, is appointed as responsible for management of the location information for all nodes in the area. The location databases in the responsible nodes can be seen as analogous to the visitor location registers (VLRs) in cellular networks. Since nodes in ad hoc networks may be very prone to failures, one or several backup nodes may be defined for each area in order to increase robustness. If node n is within the area $A^{(k)}$, then $A_n = A^{(k)}$. Accordingly, location update messages for node n are only delivered to the responsible nodes for $A^{(k)}$, say $n^{(k)}$. When needing location information for node n_2, a node n_1 located in $A^{(k)}$, for any $k \leq K$, first sends an appropriate inquiry to $n^{(k)}$. If n_2 is located in $A^{(k)}$, i.e., $A_{n_1} = A_{n_2} = A^{(k)}$, then $n^{(k)}$ is aware of the current position of n_2 and will therefore answer the inquiry sent by n_1. Otherwise, $n^{(k)}$ will send an inquiry to the responsible nodes for the other areas, i.e., $n^{(1)}, n^{(2)}, \cdots, n^{(k-1)}, n^{(k+1)}, n^{(K)}$. If n_2 is active, then it will be under the responsibility of one of the above nodes, which will answer the inquiry sent by $n^{(k)}$. This approach exploits a property that is well known in communication, i.e., *call locality*. This property is based on the fact that communications between near nodes are more frequent than communications between far nodes. In fact, this property involves that most communications occur between nodes in the same area and the responsible node is thus the same for both the caller and the called nodes. However, if this is not the case, the network may be flooded with inquiry messages between responsible nodes. To solve this problem hierarchical architectures have been proposed which divide the region into areas and subareas, e.g., [9]. The main disadvantage of this approach is that if nodes n_1 and n_2 are in different areas, i.e., $A_{n_1} \neq A_{n_2}$, a large amount of signaling may be required.

Moreover, this second approach requires each node to have knowledge of its position. This can be achieved by means of either GPS or other techniques [11]. The choice of alternative localization methods, other than GPS, is a key feature, especially in environments in which natural obstacles (like mountains or buildings) or signal weakness (as in indoor environments) do not allow the use of GPS. In addition to this, it must be observed that GPS is not appropriate due to its high cost and heavy package load which are not suitable for small highly mobile terminals.

B.2 Distance-Based Schemes

In distance-based approaches, nodes are divided into clusters. A *cluster* is a group of nodes that are close to each other for a certain period of time. One or more nodes act as representatives of each cluster and are responsible for the location information of all the nodes belonging to the cluster [12], [13]. Accordingly, in order to identify the location of each node the following information is required

- The cluster to which the node belongs. This information may be denoted as cluster membership.
- The location of the cluster representative, which is approximately equal to the location of all the nodes in the cluster (at least for small-size clusters).

Obviously, in *distance-based* schemes only the cluster representative is required to update its location information. All other nodes are only required to update the information about their cluster membership and to communicate this information to the cluster representative. This is necessary because nodes move independently and may therefore leave a given cluster and join or form a new one.

As a consequence, the following two sets of operations are required

1. Exchange of location information for the cluster representatives. This can be done using any location management scheme of the discussed above.
2. Dynamic management of cluster membership. This management must satisfy some requirements such as reliability, robustness, and energy efficiency.

As an example, in [13] a group management scheme called *MANGO* is introduced. In order to implement dynamic management of cluster membership, procedures such as *group querying*, *group joining*, *group leaving*, and *group formation* are introduced.

Cluster representatives have to perform more processing than other nodes. Consequently, a procedure for rotation of the role of cluster representative between the nodes in the cluster should be introduced as in [13]. This improves fairness because nodes in mobile ad hoc networks have generally the same capabilities.

Obviously, *distance-based* schemes are particularly convenient when nodes spontaneously move in groups, which is typical of military and disaster recovery operations. If, on the other hand, nodes frequently change membership, a large signaling overhead is required for group management and distance-based approaches are therefore not appropriate.

IV. UNSTRUCTURED SCHEMES

Fair distribution of the workload among nodes is the key advantage of *unstructured* schemes. In fact, these approaches are based on distributing location information to all network nodes without any restriction due to their identities or characteristics.

In the following sections we will discuss in greater detail the two categories of *unstructured* schemes for location management already identified in Section II.

A. Introvert Schemes

Introvert schemes are those where nodes do not spontaneously forward location information to their neighbors. On the contrary, they wait for explicit requests for location information to be sent by neighboring nodes.

More specifically, each node contains a location table with location information for certain nodes. If a node, n_1, needs to send certain data to another node, n_2, and does not have information about the position of n_2, it sends a `location request` message to its neighbors. If node n_1 does not receive an answer within a certain time interval, it floods the `location request` message to the whole network. Upon receiving a new `location request` message, a node n' behaves as follows

- If it has information about the position of n_2 in its location table, it sends a `location reply` message to n_1 containing the information about the position of n_2.
- If it does not have information about the position of n_2 in its location table, it forwards the `location request` message.

Obviously, nodes update their location table when they overhear `location reply` messages. Location information related to a certain node is removed from the location table after a certain time, since it is considered obsolete. An example of this approach is the *reactive location service* (RLS) [14].

Introvert schemes represent a simple mechanism to reduce the amount of information packets flowing through the network to keep the location information updated. In fact, the main advantage of this approach is that resource consumption is minimized because the dissemination of location information is only triggered by explicit requests. The main problem of this technique is represented by the long latency which may be experienced by n_1 before locating n_2 if there is no information available in the network about n_2's current position.

B. Extrovert Schemes

Extrovert schemes allow the latency problem to be solved. These mechanisms are based on periodic spontaneous dissemination of location information. A typical and simple example of the *extrovert* approach is the following. Each node contains a location table where location information is stored for all nodes in the network, i.e., an entry in the location table corresponds to a node in the network. Each node generates its location information by periodically calculating its position and transmits to its one-hop neighbors this information via a `location update` packet generated at rate r. The `location update` packet contains N entries taken from the location table of the node generating this packet. The choice of the N entries can follow a round robin-like approach. A node periodically receives `location update` packets from its one-hop neighbors. Upon such an event, a node updates the entry in its location table referring to the node which generated the `location update` packet. Moreover, for each of the N entries in the packet, the node checks whether the information in this entry is more recent than the information contained in its location table[1]. If this is the case, then the node updates the entry in its table; otherwise, the node ignores the entry just received. What is worth noting is that such an update of location information is executed by exchanging location table frames with neighbors, so that a complete picture of the network is available at each node. This guarantees that location information is always available when needed, without any delay. An example of this simple strategy can also be found in [14].

However, in the approach described above, two types of problems still survive. The first is that a large amount of location information is disseminated which is likely not to be used. This implies an unnecessary signaling overhead and therefore inefficient utilization of energy and bandwidth resources. The second problem is represented by the explosion in the size of the location table when the number of users in the system increases. This scalability concern is an open problem which can be reduced by combining extrovert schemes and clustered mechanisms.

Solutions to the first problem are proposed in the *distance routing effect algorithm for mobility* (DREAM) [15]. This location management scheme reduces the amount of location information being exchanged in the following way

- Location information about far nodes is exchanged less frequently than information about close nodes. In fact, the higher the distance between two nodes, the slower they appear to move with respect to each other. This is called the *distance effect*.
- The rate r at which the `location update` packets are generated, in the case of nodes moving at low velocity, is lower than the location update rate for nodes moving at high velocity. This is called the *mobility effect*.

V. CONCLUSIONS

The mobility and limited energy, processing, and storage capabilities characterizing nodes in ad hoc networks cause a need for location management approaches unlike those used in wired networks.

In this paper a taxonomy of location management strategies for mobile ad hoc networks has been provided. A classification has been introduced which distinguishes between *structured* and *unstructured* schemes, where the difference lies in whether location information is forwarded to (and/or stored by) certain nodes depending on their characteristics, or flooded to all network nodes.

Furthermore, *structured* schemes have been classified as *adaptive* and *non-adaptive* schemes. *Adaptive* schemes, which can be divided into *position-based* and *distance-based* approaches, are more appropriate when it is desirable to propagate location information only to a subset of nodes based on their position or reciprocal distance. Otherwise, *non-adaptive* schemes may be more convenient due to their simplicity, although problems of robustness and scalability may be encountered.

Unstructured schemes can be classified as *extrovert* or *introvert* schemes based on whether nodes spontaneously disseminate location information or not. *Introvert* schemes are more convenient when new calls are initiated at a low rate. If the rate is high, then *extrovert* schemes are more appropriate.

We have discussed the advantages and disadvantages of both structured and unstructured approaches and shown that

- *Unstructured* schemes allow the workload to be fairly distributed among network nodes; this may result in longer network lifetime.
- *Structured* schemes may be more appropriate in scenarios where some devices are more robust and have higher energy

---

[1] To this purpose, appropriate timestamps can be utilized.

and/or computational capabilities than others and thus higher efficiency and reliability can be achieved.

ACKNOWLEDGMENTS

This work was supported by MIUR under contract PATTERN.

REFERENCES

[1] I. F. Akyildiz *et al.*, "Mobility management in next generation wireless systems," *Proc. IEEE.* vol. 87, no. 8, Aug. 1999.

[2] I. Stojmenovic, *Location Updates for Efficient Routing in Ad Hoc Wireless Networks, Handbook of Wireless Networks and Mobile Computing.* John Wiley & Sons, 2002.

[3] Z. J. Haas and B. Liang, "Ad hoc mobility management with uniform quorum systems," *IEEE/ACM Trans. Networking.* vol. 7, no. 2, Apr. 1999.

[4] Z. J. Haas and B. Liang, "Ad hoc mobility management with randomized database groups," in *Proc. IEEE ICC'99,* June 1999.

[5] I. Stojmenovic, "A routing strategy and quorum based location update scheme for ad hoc wireless networks," *Technical Report TR-99-09,* University of Ottawa, Ontario, Canada, Sept. 1999.

[6] J. P. Hubaux *et al.*, "Towards self-organized mobile ad hoc networks: The terminodes project," *IEEE Commun. Mag.*, vol. 39, Jan. 2001.

[7] K. N. Amouris, S. Papavassiliou, and M. Li, "A position based multi-zone routing protocol for wide-area mobile ad hoc networks," in *Proc. VTC'99,* Apr. 1999.

[8] J. Li *et al.*, "A scalable location service for geographic ad hoc routing," in *Proc. 6-th Annual ACM/IEEE Mobile Computing and Networking,* Aug. 2000.

[9] I. F. Akyildiz, W. Yen, and B. Yener, "A new hierarchical routing protocol for dynamic multihop wireless networks," in *Proc. IEEE INFOCOM'97,* Apr. 1997.

[10] C. Maihofer, "A survey of geocast routing protocols," *IEEE Surveys and Tutorials.* vol. 6, no. 2, Apr. 2004.

[11] S. Capkun, M. Hamdi, and J. P. Hubaux, "GPS-free positioning in mobile ad hoc networks," in *Proc. HICSS-34,* Jan. 2001.

[12] G. Chen and I. Stojmenovic, "Clustering and routing in mobile wireless networks," *Technical Report TR-99-05,* University of Ottawa, Ontario, Canada, June 1999.

[13] L. Galluccio, G. Morabito, and S. Palazzo, "Spontaneous group management in mobile ad hoc networks," *Wireless Networks (WINET).* vol. 10, no. 4, July 2004.

[14] T. Camp, J. Boleng, and L. Wilcox, "Location information services in mobile ad hoc networks," in *Proc. IEEE ICC 2001,* June 2001.

[15] S. Basagni *et al.*, "A distance routing effect algorithm for mobility (DREAM)," in *Proc. ACM/IEEE Mobicom'98,* Oct. 1998.

Laura Galluccio received the laurea degree in Electrical Engineering from University of Catania, Italy, in 2001, and is currently working toward her Ph.D. degree with the same university. Since 2002 she is also with the Italian National Consortium of Telecommunications (CNIT), where she is working as a research fellow within the VICOM (virtual immersive communications) project. Her research interests include adhoc and sensor networks, protocols and algorithms for wireless networks, and network performance analysis.

Sergio Palazzo received his degree in electrical engineering from the University of Catania in 1977. Until 1981, he was at ITALTEL, Milano, where he was involved in the design of operating systems for electronic exchanges. He then joined CREI, which is the center of the Politecnico di Milano for research on computer networks. Since 1987 he has been at the University of Catania, where is now a Full Professor of Telecommunications Networks. In 1994, he spent the summer at the International Computer Science Institute (ICSI), Berkeley, as a Senior Visitor. He is a recipient of the 2003 Visiting Erskine Fellowship by the University of Canterbury, Christchurch, New Zealand. Since 1992, he has been serving on the Technical Program Committee of INFOCOM, the IEEE Conference on Computer Communications. He has recently been the General Vice Chair of the ACM MobiCom 2001 Conference. He has also been the General Chair of the 2001 International Tyrrhenian Workshop on Digital Communications, focused on "Evolutionary Trends of the Internet." Since 1997, he is a member of the Editorial Board of the *IEEE Wireless Commun. Mag.* (formerly the *IEEE Pers. Commun Mag*). Recently, he joined the Editorial Boards of the journals Computer Networks, Ad Hoc Networks, and Wireless Communications and Mobile Computing. He was a Guest Editor of Special Issues in the *IEEE J. Select. Areas Commun.* ("Intelligent Techniques in High-Speed Networks,") in the *IEEE Pers. Commun. Mag.* ("Adapting to Network and Client Variability in Wireless Networks,"), in the Computer Networks journal ("Broadband Satellite Systems: A Networking Perspective"). He is the recipient of the 2002 Best Editor Award for the Computer Networks journal. He also is a member of the Steering Committee of the European Virtual Center on Wireless Internet (EVC-WIN). His current research interests include mobile systems, wireless and satellite IP networks, intelligent techniques in network control, multimedia traffic modelling, and protocols for the next generation of the Internet.