A Virtual Topology for Routing in Adhoc Networks

Hiba Hachichi, Samia Chelloug(*), Fatima Athmouni
Computer science department
Mentouri University
(*)samia_chelloug@yahoo.fr

***Abstract*—Routing in adhoc networks is often solved through classical algorithms such as AODV, OLSR,... Several recent works have evaluated such algorithms and have investigated self-organization for reducing the number of messages in a dense and a mobile deployment. The main idea is to create and maintain locally a hierarchy that is well suitable for routing packets in an Adhoc network. The contribution of this work is mainly based on the construction of a virtual topology where cluster heads and gateways collaborate for searching the destination node. Hence, inter-cluster and intra-cluster routing are jointly used. The Netlogo platform has been investigated for constructing in an asynchronous manner a virtual topology. Results showed a significant reduction in the exchanging messages. By comparison to the existing methods, our contribution is also able to find the shortest path between a source and a destination.**

***Keywords*— self-organization, adhoc networks, routing, Netlogo.**

I. INDRODUCTION

Today, people move and communicate a lot, in addition, they need new technologies, enabling them to quickly and easily retrieve various types of information and communicate with distant people. So, adhoc networks have emerged to meet these new needs. They are mobile radio networks without the aid of a fixed infrastructure or a centralized administration [12, 4,1] which allow them to be deployed easily as scalable topologies.

Many routing protocols for ad-hoc networks are available. These protocols can be classified into three categories [13,18]: reactive, proactive and hybrid. The reactive [9] seek to set up routes on-demand. A node initiates a route discovery when it must send a packet and it doesn't know any route to destination. In the proactive approach [17], each node knows a route to each network node. Among the reactive protocols developed, we can find: Dynamic Source Routing (DSR), Adhoc On Demand Distance Vector (AODV), Temporally Ordered Routing Algorithm (TORA) and Associativity Based Routing (ABR). Proactive protocols have been proposed such as Destination Sequenced Distance Vector (DSDV), Wireless Routing Protocol (WRP), Global State Routing (GSR), Clusterhead Gateway Switch Routing (CGSR) et Fisheye State Routing (FSR). Hybrid protocols combine the advantages of both proactive and reactive routing protocols [7]. As an example we can cite *Zone* Routing Protocol (ZRP) [19].

However, none of these protocols can be used on networks of large scale, because they generate too much traffic control which would require too large routing tables. One solution proposed for routing on large scales is to introduce a hierarchical routing by grouping entities into clusters.

This article is organized as follows; in the next section we present self-organization and a set of self-organization structure in ad hoc networks. Then, we describe our routing solution. In section 4, we detail our application. Some simulation results will be presented to demonstrate the effectiveness of the routing protocol proposed.The final section concludes the paper and gives some perspectives.

II. SELF-ORGANIZATION

Self-organization is a set of mechanisms to produce a global and stable state of a system from interaction of different units without any interaction to the external environment.
A self-organized system is based on:

- The local interaction: the emergent behavior of the system is more than simple interactions between its various elements without external control.
- The emergence of a global structure: the main objective of self-organization is to produce

stable structures which are constructed in coherently bounded time [16, 6].

- Adaptation to the environment and robustness: the emerging structure of a system must be adjusted to the environment and respond to local changes [2, 11].
- Large scale: this property is the result of the absence of central control and internal interactions.

A. Self-organization in ad hoc networks

Among the structures which have served to the self-organization of an Ad hoc network, we can find:

- *Backbone:* is a network that concentrates traffic from other networks in a hierarchical structure to ensure full communication [16]. Only the backbone nodes are allowed to relay a broadcast traffic. Any node must be near the backbone for sending it the information to be disseminated [8,14].
- *Cluster:* the zoning of an extended network allows organizing it for addressing problems of routing and aggregation of flows. If a head is elected in each zone, a hierarchy is created.

1. Distributed algorithms to construct a self-organization structure

The topology control in ad hoc networks is a recent domain of research. It aims to maintain an adequate topology by controlling the links to be included in the network. Valois and theoleyre [16, 15] proposed an algorithm which is based on the following steps: acquisition of neighborhood knowledge, construction of a virtual backbone and clusters, and maintenance of the virtual structure. Mitton[10] introduced a new metric called density. Her algorithm is divided into three phases: the computing of the density, designation of clusterheads, construction of clusters and maintenance of the virtual structure. In [5] Ferjani and Ayari presented an algorithm of clustring named « distribution reaffiliation controlled clustring algorithm ». It allows forming clusters dynamically with variable radius. Cluserheads are chosen on the basis of some criterions and the stability of clusters formed is assured through the introduction of constraints on reafilliation.

The main idea of the algorithm Drira, Khadoussi and Tabbane: [3] is to construct a set of dominants nodes in the network. The dominant character of such algorithm is the number of diffusion path to reach nodes of two jumps. Fixation of dominants nodes represents an essential step for the construction of set of clusters whose diameter is greater than four. Also, each cluster is identified by three categories of nodes:

(1) The cluster-head (CH): is a dominating node and is the cluster chef.
(2) The secondary cluster-heads (CHS): is a subset to CH dominants nodes to CH. These nodes agree to the same CH.
(3) Ordinary nodes (NO) are not dominants nodes.

CH acquires total knowledge of its cluster nodes. Each request is broadcasted and passes through the CH. If the destination is not founded in the neighbors table (its neighbors and their neighbors), the request is therefore transmitted to the CHSs which circulate it to the gateways nodes. A gateway allows disseminating information to neighboring clusters. A cluster maintains a tree structure useful for eliminating loops in the cluster. Thus, the network itself may be considered as a set of related zones.

The principle of the algorithm is summarized as follows:

Initially, each node in the network selects its preferred node (also called broadcast node): the node which ensures a better distribution in two jumps. Then, the broadcast node having the higher choice average declares itself as CH. The choice average is the ratio of the number of dominants nodes which are neighbors to a node *u* and have chosen the same node *u* as a broadcast node divided by the number of neighbors. For NOs, they are always attached to the broadcast node. These nodes have a null choice average. A forest is constructed by relating the CHs, the CHSs and NO*s*. Then, the algorithm constructs the routing table intra Cluster to provide an appropriate structure for each tree. This one is located only at the CH. For CHSs and NOs, they maintain a vector (next node to receive information) to their broadcast node. Also, the algorithm constructs a table inter Cluster in gateways. It should be noted that each Cluster has a unique identifier which is the identity of the CH.
The different phases are established according to the information provided by messages HELLO which form a vector structure, HELLO= {ZID, VID, N, DEG, BP, BN, AVRG, GTWY}.

III. CONCEPTION OF ROUTING PROTOCOL

This section describes our routing algorithm based on the topology proposed by Drira, Khadoussi and Tabbane [3]
The Hello message format is as follows:

- Zidttnoeud (ZID); Cluster identity.
- Idnoeud (VID); the identifier of the node.
- Neighbor (N); list of neighbors nodes.
- NB (DEG) ; the number of neighbors.
- BP; the broadcast parameter of a node, is the set of all possible paths to reach its neighbors of two jumps.
- ND or BN; broadcast node, the preferred node (having the maximum value of BP) for the distribution to two jumps.
- Average (AVRG) ; The choice average of a BN.
- Neighborgateway; to indicate the nodes which are gateways.

This algorithm is based on a hybrid protocol which is summarized as follows: (figure 1):

- Inside of a cluster, a proactive routing protocol is adopted.
- A reactive protocol is conceived inter-cluster.

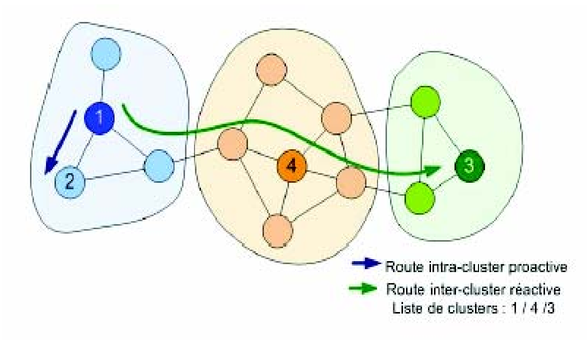


Figure 1. General description of the routing protocol

To find the destination, the source sends a request "route_request", which forms a structure of a list:
route_request = {Source, Intermediate, Destination, Dis } with:

- Source: contains the identity of the source node.
- Intermediate: Indicates the identities of intermediate nodes. Each node saves its identity before broadcasting the request.
- Destination: contains the identity of the destination node.
- Dis: indicates the distance between each intermediate node and the source.

Once the destination receives the 'route_request' request, it sends a route_reply response which has almost the same structure as the 'route_request' except it doesn't use the last field **Dis.** This response follows the reverse path recorded in the destination node.

A. Phase of intra Cluster routing

This phase allows a source node to find the destination within the Cluster.

The source verifies if the destination exists in its neighborhood of one or two jumps:

- If so, the source will choose the shortest path leading to this destination.
- Otherwise it will send a 'route_request' packet to its broadcast node as it passes t to its Cluster Head (*CH*). This later verifies the presence of the destination in its cluster and informs the source by its existence or not.
- If the destination doesn't exist in the cluster, the source initiates a search in the other clusters. (Phase of inter cluster routing).

B. Phase of inter cluster routing

Routes between clusters are discovered by a reactive routing protocol to fully exploit the network hierarchy.
When the source wants to send a packet to its destination which is not in its cluster, the following cases may occur:

- A route to the destination is present in the routing table of the source: it directly executes its algorithm of inter cluster routing to reach the destination.
- Unknown destination: the source initiates a 'route_request' to its neighbors"clusters Head or broadcast nodes"(ordinary nodes are no longer part of this phase).
- If the source is an ordinary node, it sent directly the 'route_request' to his broadcast node which generates a route discovery.

The clusters Head or the broadcast nodes which receive a 'route_request' save their identities in the field Intermediate and their distances in the field Dis ; then they test if the destination exists in their neighborhood; otherwise, they send the new request to gateway nodes(Cluster Head "CH" or broadcast nodes "ND") consulting their inter cluster routing table.
Once the 'route_request ' requests arrives at the destination, it chooses the shortest route and sends a 'route_reply' response which will follow the reverse path to inform the source node of the complete path to the destination.

Our algorithm has the following advantages:

- Calculate all possible routes leading from a source to a destination.
- Optimize the number of messages circulating in the network:
  - The source itself generates the discovery of the road.
  - Ordinary nodes are no longer part of gateways.
- To avoid the problem of loops:

- Each gateway sends a single packet that came from the same node.

IV. SIMULATION RESULTS

We choose the simulation environment Netlogo to implement our conception. It allows parallel and dynamic simulations. The realization consists of two steps:

- Construction of topology
- Implementation of the routing protocol

A. Construction of the self-organization topology

First we must create an ad hoc network with random mobility (figure 2), the button setup allows to collect the network.
To vary the number of nodes in the network using the slider number-of-nodes.

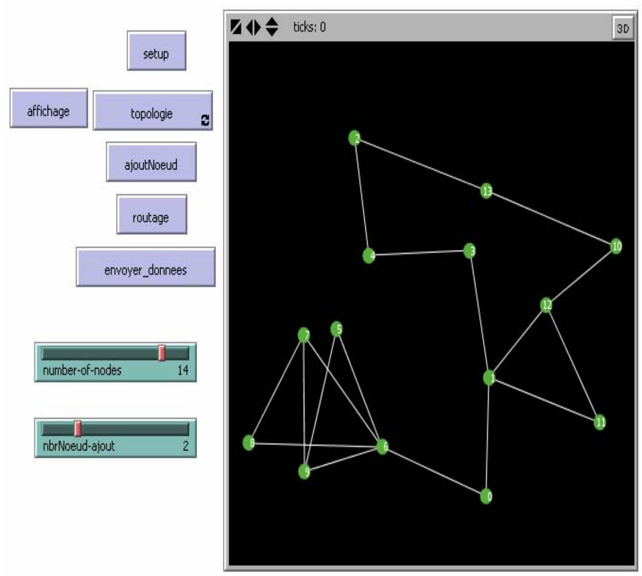


Figure 2. Network construction

By pressing the button ‘ topologie’, our topology is constructed (figure 3), ‘ajoutNoeud’ button is used to maintain the topological changes of the network.

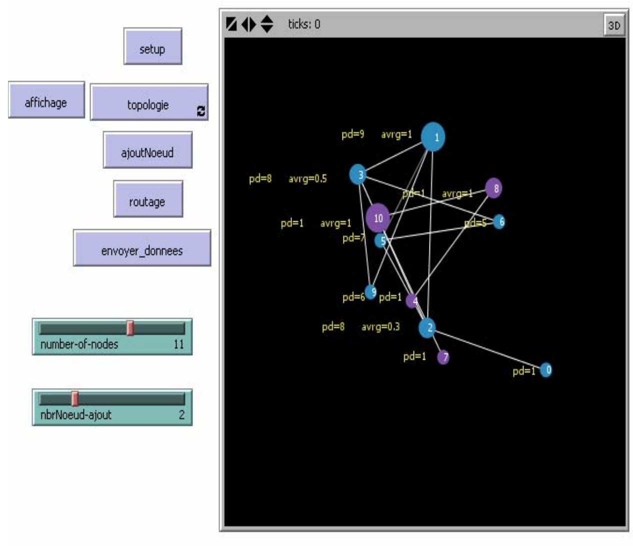
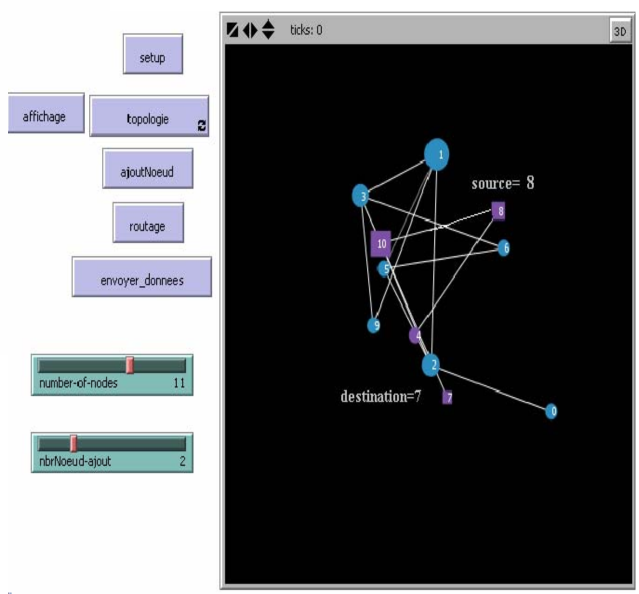


Figure 3. Construction of self-organization topology

B. Implementation of the routing protocol

Pressing on the button ‘routage’ (Figure 4), the source begins looking for its destination. (Phase of intra Cluster routing (figure 4) and Phase of inter cluster routing (figure 5)).



observer: "la route la plus courte:"
observer: [8 10 7]
observer: "la distance de la route la plus courte: "
observer: 30.75

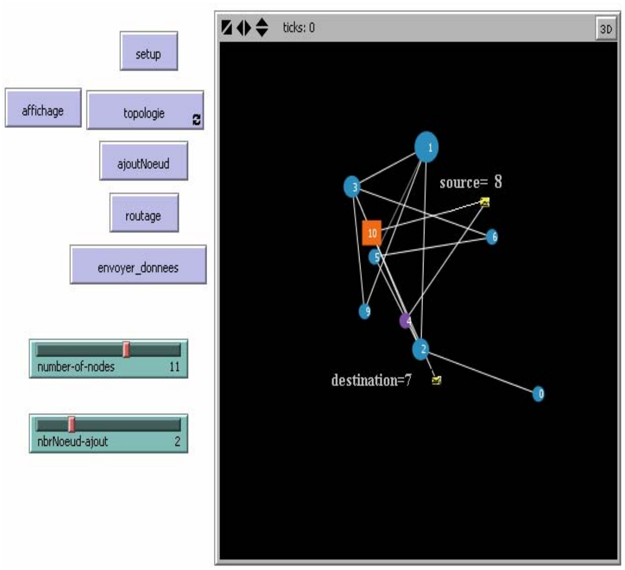
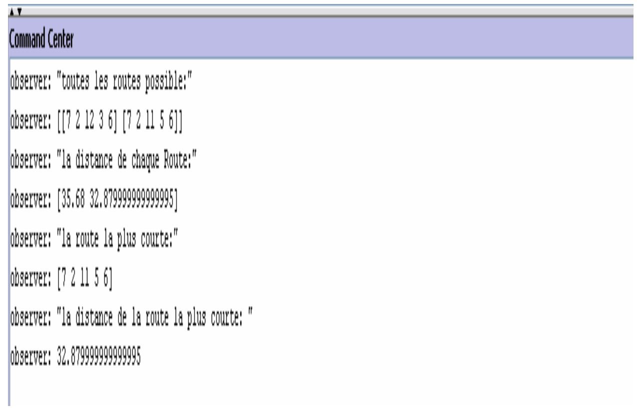


Figure 4. Phase of intra Cluster routing



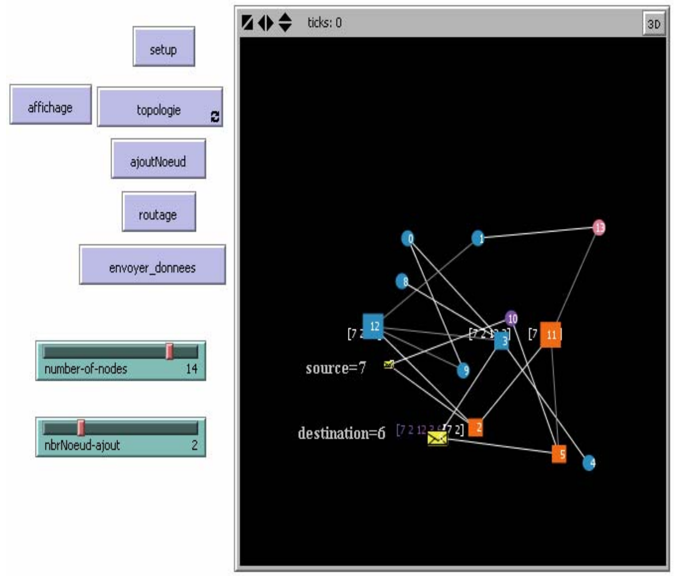
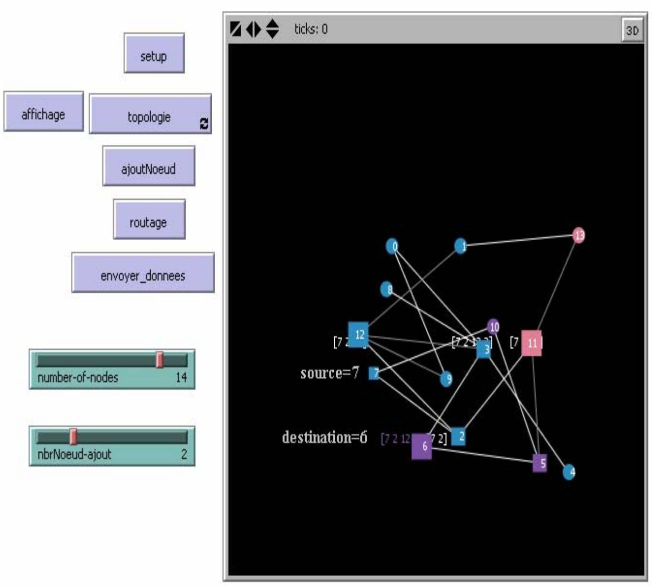


Figure 5. Phase of inter cluster routing

Source nodes, destination and intermediaries take the form of a square. When the shortest route to the destination is discovered, the source sends its data using the ‘envoyer_donnees’ button (figure 4) and takes the form of an envelope. The intermediate nodes become orange. In the reception of data, the destination takes form of envelope.



C. Results analysis

To demonstrate the efficacy of our proposition, a comparison was made between our routing algorithm (algo1 **:** figure 6**)** and the algorithm proposed by [3] **(**algo2 **:** figure 6).
The purpose of this comparison is to show that our proposition reduces more the number of messages propagated in the network what really reflects the figure 6, such we note that the curve representing algo2 is higher than that which represents algo1. This justifies the objective of the implementation structure.

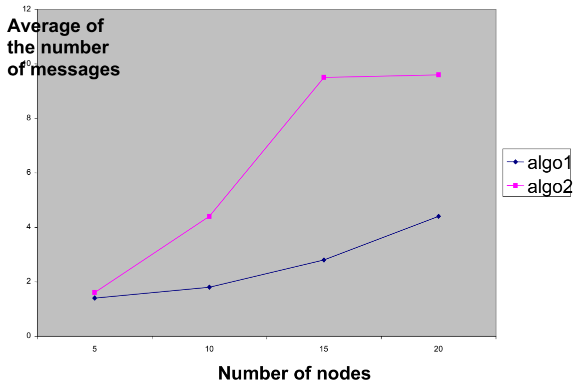


Figure 6. Comparison between algo1 and algo2

V. CONCLUSION AND PERSPECTIVES

In this paper, we proposed a distributed algorithm for routing in mobile ad hoc networks based on a self-organization structure by exploiting a hybrid routing protocol. It has the advantage of reducing the number of messages propagated in the network and finding the shortest path. To prove these advantages were introduced a simulation which compares our proposed routing algorithm to another algorithm.

Having proposed this work, several perspectives are worth noting:

- How self-organization conceived for ad-hoc networks can be adapted for sensor networks?
- It is desirable to think of a more sophisticated programming environment which facilitates the simulation of roads between the layers of network.

REFERENCES

[1]: S. Basagni, Distributed clustering for ad hoc networks, in: Proceedings of International Symposium on Parallel Architectures, Algorithms and Networks, June 1999, pp. 310–315.

[2] :M. Chemilier. Les mathématiques naturelles. Odile Jackobe coll ‘sience’, 2007.

[3] : k. DRIRA , H.KHEDDOUCI ,N. TABBANE , Topologie dynamique virtuelle pour le routage dans les réseaux mobiles Ad hoc, SETIT'2007, Tunisie, pp. ISBN 978-9973-61-474-2 (2007).

[4] : E.Fleury, Communication de groupe du parallélisme à l’ad hoc. Université Claude Bernard - Lyon 1, 2002.

[5] :H. Ferjani, M. Ayari, DRC: Mécanisme de clustering pour la gestion par politiques dans les réseaux ad hoc, Colloque francophone sur l’Ingénierie des Protocoles, CFIP 2006, novembre 2006, Tozeur, Tunisie. (short paper).

[6] : J. George. Résolution de problèmes par émergence. Université Toulouse - Paul Sabatier, 2004.

[7]: V. Gayraud, L. Nuaymi , S. Gombault ,B. Theron. La Sécurité dans les Réseaux Sans Fil Ad Hoc. Proc. of SSTIC (Symposium sur la sécurité des technologies de l'information et de la communication), Rennes, France, jun. 2003.

[8]: B. Liang and Z. J. Haas, Virtual backbone generation and maintenance in ad hoc network mobility management, in Proc. IEEE INFOCOM, March 2000, pp. 1293 – 1302

[9]: M.Mehdi, A.Anou, S.Zair. La Sécurité dans les réseaux Ad Hoc, 4th International Conference: **S**ciences of Electronic,Technologies of Information and Telecommunications ,SETIT 200**7,** March 25-29, 2007 ,TUNISIA.

[10] : N.Mitton. Auto-organisation des réseaux sans fil multi sauts à grande echelle. Lyon, 2006.

[11]:C. Prehofer, C. Bettstetter. Self- organisation in communication net works: Principles and design paradigms, IEEE Communications Magazine, Vol. 43, No. 7. (July 2005), pp. 78-85.

[12] :V. Paruchuri, A. Durresi, D. Dash, R. Jain, Optimal Flooding Protocol for Routing in Ad-hoc Networks, Submitted to IEEE Wireless Communications and Networking Conference (WCNC 2003), New Orleans, Louisiana, March 16-23, 2003.

[13]: E.M. Royer, C-K. Toh, A Review of Current Routing Protocols for Ad hoc Mobile Wireless Networks. IEEE Personal Communications, Vol. **6,** No**. 2,** Pages **46-55, April 1999.**

[14]:A. Schumacher, “Dominating set based routing,” in. Algorithms for ad hoc networking, 2003, pp. 1-4.

[15] : F.Theoleyre. Une auto-organisation et ses applications pour les réseaux ad hoc et hybrids. Lion, 2006.

[16] : F. Valois. Auto-organisation de réseaux radio multi-sauts. Université Claude Bernard, Lyon 1, 2007.

[17] : X.Xue. Mécanisme de sécurité pour des protocoles de routage des réseaux ad hoc. Ecole nationale supérieure des télécommunications, Paris, 2006.

[18]: S. Zhao, K. Tepe, I. Seskar and D. Raychaudhuri, "Routing protocols for self-organizing hierarchical ad hoc wireless networks," IEEE Sarnoff Symposium, Trenton, NJ, March 2003.

[19]: X. Zou, B. Ramamurthy and S. Magliveras. Routing Techniques in Wireless Ad Hoc Networks– Classification and Comparison. Proceedings of the Sixth World Multiconference on Systemics, Cybernetics and Informatics, SCI. 2002