Computer Science Journal of Moldova, vol.18, no.3(54), 2010

Service for fault tolerance in the Ad Hoc Networks based on Multi Agent Systems *

Ghalem Belalem, Esma Insaf Djebbar, Abderahmann Benaissa, Ali Cherif Brakeche

Abstract

The Ad hoc networks are distributed networks, self-organized and does not require infrastructure. In such network, mobile infrastructures are subject of disconnections. This situation may concern a voluntary or involuntary disconnection of nodes caused by the high mobility in the Ad hoc network. In these problems we are trying through this work to contribute to solving these problems in order to ensure continuous service by proposing our service for faults tolerance based on Multi Agent Systems (MAS), which predict a problem and decision making in relation to critical nodes.

Our work contributes to study the prediction of voluntary and involuntary disconnections in the Ad hoc network; therefore we propose our service for faults tolerance that allows for effective distribution of information in the Network by selecting some objects of the network to be duplicates of information.

Keywords: Ad Hoc network, Fault tolerance, Disconnection, Replication, MAS.

1 Introduction

The mobile Ad-Hoc, usually called MANET, are wireless networks, mobile, spontaneous, composed of a set of nodes grouped dynamically and are usually connected by radio waves, not based on any existing infrastructure or centralized administration. The nodes are free to

* This work was supported by Laboratory LIO (Es Senia) Oran, Algeria

move randomly and organize themselves arbitrarily; thus, the topology of the wireless network may change rapidly and unpredictably. Such a network can operate in standalone mode, or be connected to the Internet. An Ad hoc network may consist of several devices such as PDAs, laptops, etc. Each node can communicate directly with other nodes residing in its transmission range. To communicate with nodes that do not live around its radius, the node has to use intermediate nodes to send the message per hop.

In wireless networks, especially ad hoc networks have new problems, specific to mobile environments, such as management of disconnections that result in a loss of information. For the fault tolerance, the replication is a means to ensure continuity of service for data sharing in the network. The disconnections are voluntary or involuntary. The first disconnection is decided by user from his mobile terminal. The latter are result of falling energy nodes; partition of the network, or the frequency of simultaneous failures of nodes.

The objective of this work is to build a service of fault tolerance by replication in Ad hoc Network that incorporates the features necessary for improved data availability. Our solution is based on the cluster formation to better manage the network, and each is managed by a leader node. This solution is mainly composed of four sub-services, namely clustering, decision, replication, and consistency.

The rest of the paper is organized as follows. After the literature review of the Ad hoc networks in Section 2, we discuss the approaches of faults tolerance in Section 3, and the related works in Section 4. In Section 5, we specify our proposal service of fault tolerance by detailing these various sub-services. Then we present our various experiments conducted by our simulator developed in Java. Finally, we conclude and give perspectives of our work.

2 Ad hoc Network

A mobile Ad-Hoc network (MANET) is a group of mobile nodes forming a temporary network without the aid of any fixed infrastructure or centralized administration [7]. The configuration of this network can be static or dynamic. Its life is variable but can be very limited. In addition, it incorporates the specific characteristics of any standard mobile network, among others [7, 10]:

- Mobility is the rule rather than the exception: it is the essence itself of the network;
- The disconnection: voluntary or involuntary, temporary or permanent;
- Instability in the storage of data: is it available at all times for backup operations?
- The low-capacity storage media except some hosts such as laptops;
- Low bandwidth (in the present state of technological progress). This factor is closely linked to technological change;
- The low power mobile sites: batteries are very far from the stability of the energy made available in fixed networks. This factor will play a very important role and is closely related to other factors;
- The limited computing power (except for laptops);
- High vulnerability to failures due to the fragility of the environment;
- The high vulnerability to intrusion.

In general, routes between nodes in an Ad hoc network may include multiple hops and, therefore, it is appropriate to call such networks *multi-hop "Ad hoc" networks*. Figure 1 shows an example of mobile Ad hoc and communication topology.

As it is shown in Figure 1, an Ad hoc network may consist of several computer devices such as PDA's, laptops and so on. Each node can communicate directly with other nodes that lie in its transmission radius. To communicate with nodes that reside around its radius, the

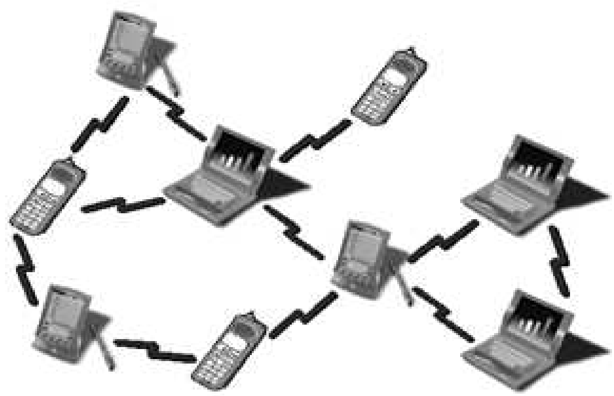

Figure 1. Ad hoc Network

node needs to use intermediate nodes to send the message by jumping. MANET can be modeled by a graph $G_t = (V_t, E_t)$ with V_t – all network nodes and E_t – all connections between these nodes. If u and v belong to V_t and $e = (u, v)$ belongs to E_t, then the nodes u and v can communicate directly at the instant t [10].

In mobile networks disconnections are voluntary or involuntary (Figure 2): the voluntary disconnections are decided by the user from his mobile terminal. Involuntary disconnection can be cut by the physical network connection, or the exhaustion of the battery, or the partitioning of the network or the failure of nodes.

3 The Approaches of Fault Tolerance

Fault tolerance is the ability of a system's performance in spite of faults [3, 11]. Other means of dependability can be considered to build secure operating systems (i.e. the elimination, prevention or prediction errors). The purpose of fault tolerance is to increase the system reliability. As part of the work in this manuscript, we focus exclusively on fault tolerance to make data availability. The requirement of fault tol-

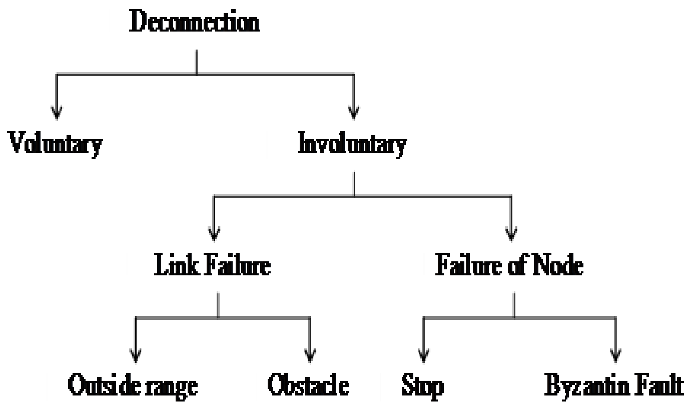

Figure 2. Disconnection in mobile environment

erance is a key player in distributed systems [4, 9]. To tolerate faults, replication is a fundamental technique used in distributed systems. It is to store the same data or service in several nodes. Data or services are often replicated to improve availability, reliability, tolerance to faults, and performance. Replication can also improve data availability and service when the server crashes. Mobility or failure of one node can lead to network partitioning, where the network is divided into disjoint partitions. This is caused by the possibility of inconsistent data.

4 Related Works

Hauspie and Buttyan [2, 5] have proposed a new metric to detect the partitioning of the network without using GPS. The metric is based on finding a set of paths between a disjoint client node and a server node. A set of paths is a disjoint set of paths that have no common node except the client node and server node. The decision to replicate a service or data is taken when the connection between the client and server is getting worse in terms of sociability, bandwidth, delay, etc. Replicating the service or the data on a node that is closest to the client node can increase the quality of the connection between the client and server nodes.

Jorgic et al. [6] proposed algorithms to detect objects located critics, such as nodes or links. A node is called k-critical if the failure of a node subgraph of neighbors of k jumps, generates a partitioning of network.

The authors in [13] have proposed a schema of replication, called the replication ring expansion (Expanding Ring Replication). The data server measures the frequency of requests for each data. If it exceeds a threshold value, they say the data on one or more nodes are capable of their neighborhood. The capacity function considers parameters, such as the available memory space, the remaining battery power and processing capacity.

5 Service for Fault Tolerance

This present work is an extension of the fault tolerance approach described in [1], where we have integrated a multi-agent system for negotiation and monitoring of each sub-service that has made our approach.

The proposed approach for fault tolerance in the Ad hoc Networks is composed of four sub services, which are presented in Figure 3. The service for fault tolerance is built with an architecture, based on clusters that are identified by a particular node, called the leader, and to each leader there is assigned a leader agent which handles itself the other three launched agents to cooperate together: recording, replication and consistency in groups for the purpose of cooperation. The model is based essentially on the concept of replication after predicting eventual disconnections or partitioning objects to be critical in the network.

5.1 Clustering sub-service

The clustering is a virtual network division in groups of nodes geographically near. These groups are called clusters. To form clusters,

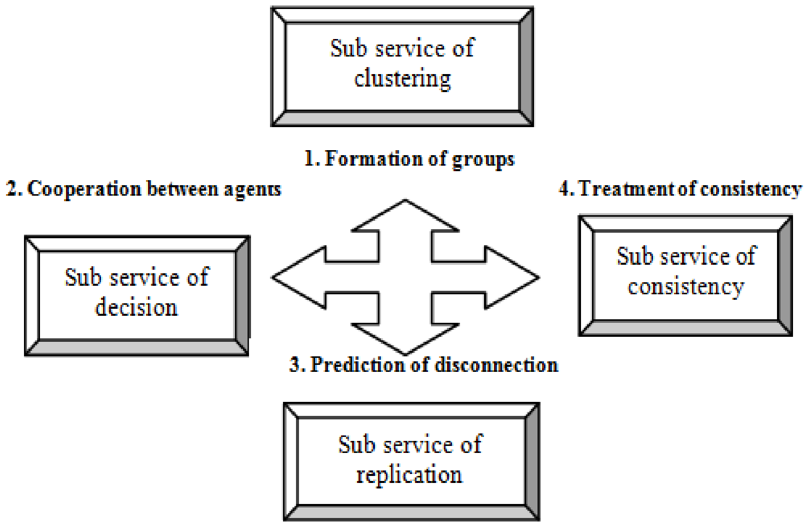

Figure 3. Service for fault tolerance

we use a distributed algorithm based on the distance between nodes and the energy level. The clustering algorithm illustrates the pseudo code for the construction of the various clusters that make up the network (see Algorithm 1).

Algorithm 1 LEADER ELECTION

```
BroadcastElectionMessage(MyID,EnergyLevel); /*broadcast identifiant and energy level to 1-hop neighbor*/
ReceiveElectionMessage(ID,EnergieLevel);
SortList(ID,EnergieLevel);
N ← FirstElementList()
if MyID = N then
    WhoIsLeader ← N;
    IsLeader ← true
else
    Father ← N
    SendElectionMessageToFather() /*for inform node chosen as father*/
    ReceptionElectionMessageFromFater() /*for know node who chose as father*/
end if
if (IsLeader = false) and (IsLeaderr(Father) = false) and (is more than 2-hops from leader) then
    Snag to a leader among the neighbors
end if
```

5.2 Decision sub-service

After the cluster construction in network, we have combined efforts of a set of agents with intelligent behavior and coordinated their goals and action plans to solve a problem. In our work, we have implemented several agents presented in Figure 4.

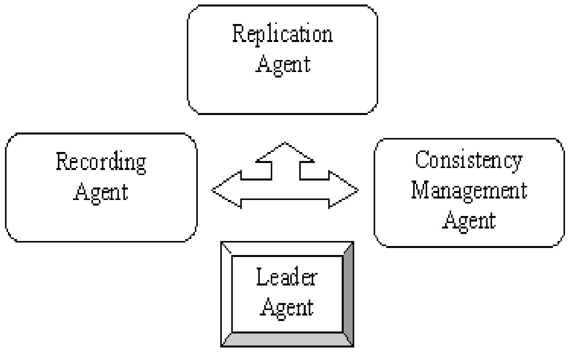

Figure 4. Different Agents implemented

The leader agent is the manager agent in the group that launches three agents; the recording agent has a charge of registering the nodes which form the group, the new nodes, and those which leave the group; the replication agent ensures the control of number of replicas in the group, following the detection of a possible disconnection in the network; and the consistency agent initiates the update propagation to nodes. The generic agent responsible for managing all the groups, it is considered the super agent, is reserved to gather information from all the leaders in the network in the case when a leader fails, and then the generic agent may substitute it to accomplish its tasks. Figure 5 shows the diagram of class UML of the various entities comprised in the system.

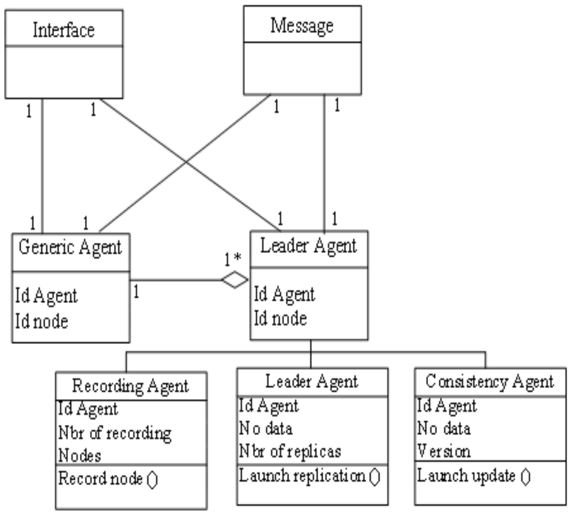

Figure 5. Diagram of class for sub service of decision

5.3 Prediction and replication sub-service

The main function of this phase is to predict a possible network failure or disconnection by establishing a list of critical (sensitive) items. Each leader can know the current status of all cluster nodes. To assess the criticality, the sub service can detect several types of critical items: energy, articulation point, Isthmus, frequency of breakdowns. Replication allows to create a full copy (or several) of an object or an object fragment. It is a basic method which allows service reliability and continuity. When the leader agent detects a possible failure or disconnection in its group, it launches the replication agent; its role is to replicate the data of the critical object detected in the node that has the highest degree of energy.

5.4 Consistency sub-service

This sub-service can manage the consistency management among replicas. The consistency management has to provide copies of their mutual consistency; all data copies are identical. In this work, we focus on the strong consistency among the replicas; the consistency agent is responsible for the update propagation when a node makes a write request by the dynamic quorum. The latter allows to write on the majority of the counterparts in each cluster and reading on half of the counterparts.

6 Simulation and Result

In this section, we study the performance of the proposed service using *AdHocFTSim*. AdHocFTSim is a simulator for Ad-Hoc network that we have developed in java. In the simulator the nodes move according to the waypoint mobility model. In this model, a node randomly selects a location and moves toward this model with a constant speed uniformly distributed between zero and a maximum speed $Vmax$; then, it stays stationary during a pause time before moving to a new random location. Initially, each mobile node has a level of energy represented in percentages. In all simulations, nodes start with full battery.

6.1 Accepted and lost requests

In this first series of experiments, we have measured the number of accepted and lost requests. This simulation has been conducted with the parameters of simulation as follows: 50 nodes, 20 data, given the size of 100 MB, bandwidth 11 Mb/s, surface of simulation 700m X 700m, the range 200 m, time of simulation 60 s, the results of this simulation are shown in Figure 6.

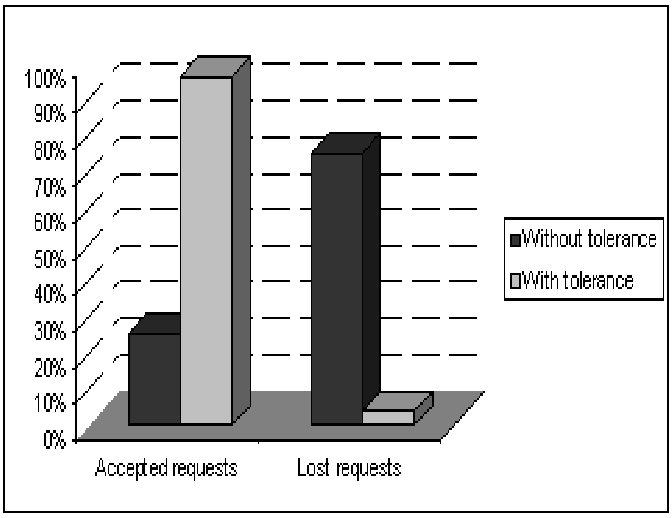

Figure 6. Accepted and lost requests

We can see that the results of our proposal are very significant, and that the number of lost requests decreases by using our service of faults tolerance.

6.2 Life of the nodes in the network

In this second series of experiments, we have measured the number of nodes alive and those failed after the simulation. This simulation has been conducted with the following simulation parameters: 50 nodes, 20 data, given the size of 100 MB, bandwidth 11 Mb/s, surface of simulation 700m X 700m, the range is 200 m, time of simulation 60 s, and number of requests 200, the simulation results are shown in Figure 7.

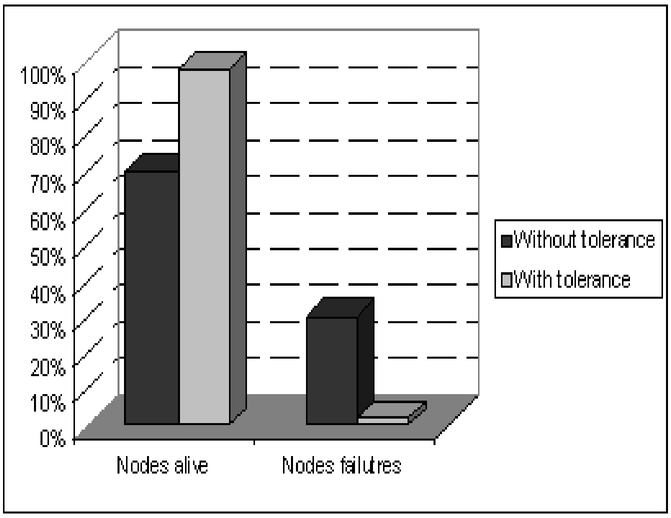

Figure 7. Life of the nodes in the Network

We can see that the results of our proposal are very significant, and that the number of nodes alive increases by using our service of fault tolerance.

6.3 Variation of number of nodes/ # lost requests

The third set of simulation results shows the effect of the node variation on the average results of lost requests. For this simulation, we vary the number of nodes from 50 to 300 in step of 50, with the number of requests = 200. By varying the number of nodes, we can demonstrate the variation of the suggested service of faults tolerance (see Figure 8 describing graphical representation of the result). We can see that the results of our proposal are very significant, and that the number of lost requests decreases by using our service of faults tolerance (see Table 1).

Table 1. Variation of the number of lost requests/ #Number of nodes

#Nodes	Without tolerance (%)	With tolerance (%)
50	27,5	1
100	30	0
150	25,5	0
200	28	1
250	28	0
300	29	0

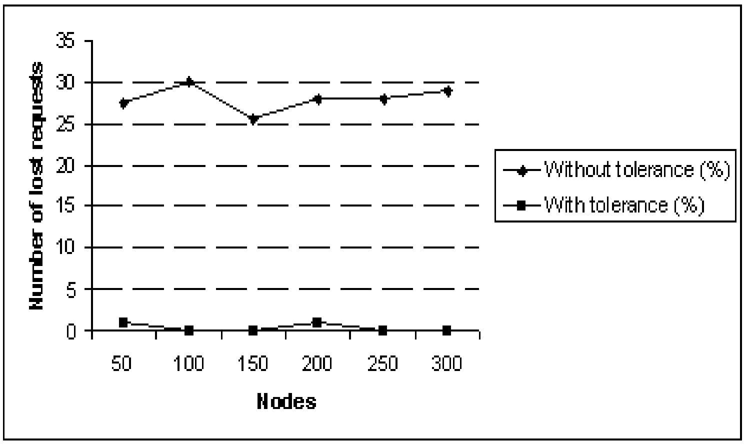

Figure 8. Variation of number of lost requests/ #Number of nodes

6.4 Variation of the number of nodes alive in the network/ #Number of nodes

The fourth set of simulation results shows the effect of variation in the number of nodes on the average results of the remainder nodes alive. For this simulation, we vary the number of nodes from 50 to 300 in step of 50, with the number of requests = 200. By varying number of nodes, we can demonstrate the variation of the suggested service of faults tolerance (see Figure 9 describing graphical representation of the result). We can see that the results of our proposal are very significant, and that the number of nodes alive increases by using our service of faults tolerance (see Table 2).

Table 2. Variation of the number of nodes alive in the network/ #Number of nodes

#Nodes	Without tolerance (%)	With tolerance (%)
50	74	88
100	87	97
150	92	100
200	93	98,5
250	94,4	100
300	95,33	99,33

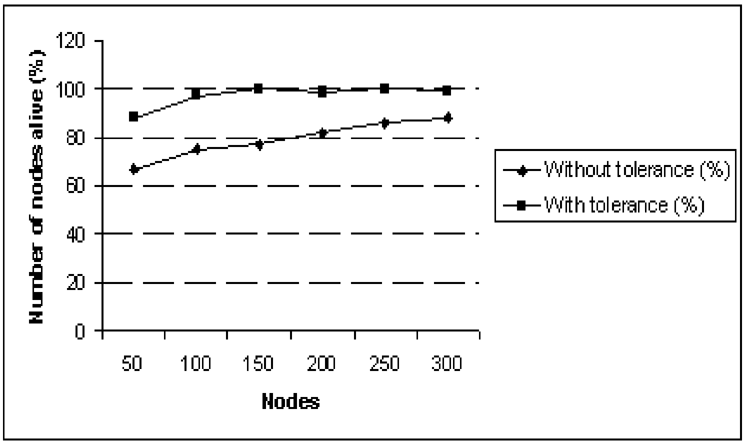

Figure 9. Variation of the number of nodes alive in the network/ #Number of nodes

6.5 Energy consumption

This series of experimentation shows the results of simulation of energy consumption. We have realized this simulation with the following parameters: 100 nodes, range 200 m, surface 700 m X 700 m, time of simulation 20 s, number of data 20, number of requests 50.

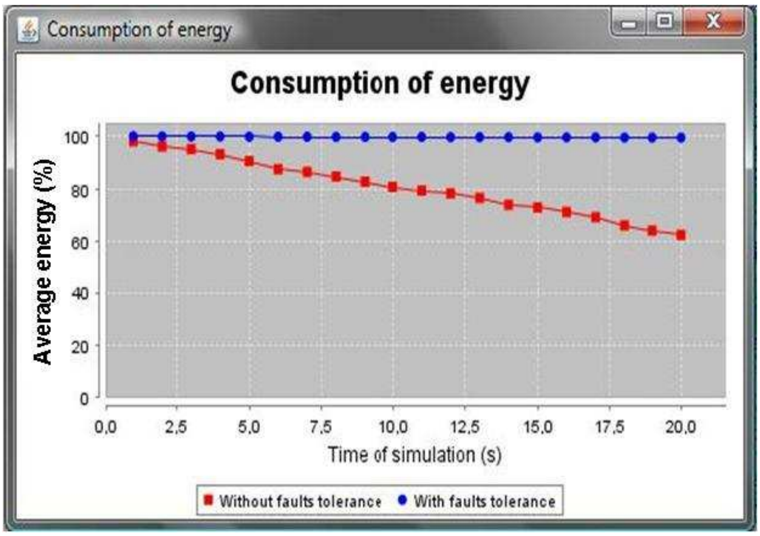

Figure 10. Energy Consumption

We notice from Figure 10 that the energy conservation with our service for faults tolerance is less expensive compared to the approach without fault tolerance that is explained by the use of clustering use which minimizes the number of sent messages. And consequently, the network life with our proposal increases significantly.

7 Conclusion and Future Works

Fault tolerance in the Ad hoc networks is a difficult problem to solve. We have presented in this paper a service of fault tolerance that includes a modeling algorithm in group in the Ad hoc networks and then apply the fault tolerance by replication of data of nodes that are critical, a fact which is essentially based on the prediction.

The defined algorithm forms groups based on geographical criteria and energy nodes. The network after clustering has a hierarchical structure of level two with a leader for each group and super leader for the network totality. The groups are open, dynamic, mutually exclusive, explicit, allowing point to point communication and group.

The applied service of fault tolerance consists of four sub-services (namely clustering, decision, replication and consistency management), to better manage the network and incorporates the features necessary for improved data availability. Our contribution takes into account the characteristics of minimizing information loss. For perspective: we intend to integrate in our simulator different protocols of routing in Ad hoc network and implement our service of fault tolerance in a real simulator for Ad hoc network as $NS-2$ or $GloMoSim$.

References

[1] G. Belalem, A-C. Brakeche, A. Benaissa, E-I. Djebbar *AD-HOCFTSIM: A Simulator of Fault Tolerence In the AD-HOC Networks.* International Journal of Wireless & Mobile Networks (IJWMN), 2(4): 159–169, 2010.

[2] L. Buttyan, J.-P. Hubaux *Enforcing service availability in mobile ad-hoc WANs.* In Proc. of the 1st ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2000). Boston, Massachusetts: ACM, (2000), pp. 87–96.

[3] B. Goncalves, N. Mitton, I. Gurin-Lassous, *Comparison of two Self-Organization and Hierarchical Routing Protocols for Ad Hoc Networks.* In Second International Conference on Mobile Ad Hoc and Sensor Networks, Hong-Kong, China, December (2006).

[4] M. Hamdy, B. Konig-Ries, *A service distribution protocol for mobile ad hoc networks.* ICPS '08: Proceedings of the 5th interna-

tional conference on Pervasive services, Sorrento, Italy (2008), pp. 141–146.

[5] M. Hauspie, D. Simplot, J. and Carle, *Partition Detection in Mobile Ad-hoc Networks.* In Proceeding of the 2nd Mediterranean Workshop on Ad Hoc Networks (Med–Hoc–Net 2003), Mahdia, Tunisia (2003).

[6] M. Jorgic, I. Stojmenovic, M. Hauspie, D. Simplot-Ryl, *Localized algorithms for detection of critical nodes and links for connectivity in ad hoc networks.* In Proceeding of The Third Annual Mediterranean Ad Hoc Networking Workshop, Med-Hoc-Net, June, Bodrum, Turkey. (2004).

[7] P. Michiardi, R. Molva, *Mobile Ad Hoc Networking.* Wiley-IEEE Press, 2004, ch. Ad Hoc Network Security (2004), pp. 329–354.

[8] N. Mitton, A. Busson, E. Fleury, *Efficient broadcasting in self–organizing sensor networks.* International Journal of Distributed Sensor Networks (IJDSN), 2(2): 161–187, 2006.

[9] P. Padmanabhan, L. Gruenwald, A. Vallur, M. Atiquzzaman, *A survey of data replication techniques for mobile ad hoc network databases.* The VLDB Journal, 17(5): 1143–1164, 2008.

[10] C. Perkins, *Ad Hoc Networking.* Addison–Wesley, 2001.

[11] B. Randell, A. Avizienis, J–C. Laprie, C. Landwehr, *Basic concepts and taxonomy of dependable and secure computing.* IEEE Transactions on Dependable and Secure Computing, 1(1): 11–33, 2004.

[12] D, Simplot, M. Hauspie, M. Carle, *Replication decision algorithm based on link evaluation for services in MANET.* Technical Report 2002-05, IRCICA/LIFL, Univ. Lille 1, 2002.

[13] V. Thanedar, K. C.Almeroth, E. M. Belding-Royer, *A Lightweight Content Replication Scheme for Mobile Ad Hoc Environments.* In

Proceeding Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communication, Third International IFIP–TC6, Athens, Greece, May 9–14, LNCS Volume 3042, pp. 125–136, 2004.

Ghalem Belalem, Esma Insaf Djebbar
Abderahmann Benaissa, Ali Cherif Brakeche

Received May 20, 2010
Revised December 16, 2011

Ghalem Belalem
Dept. of Computer Science, Faculty of Sciences,
University of Oran, Algeria
BP. 1524, El M'Naouer, Oran, Algeria (31000)
Phone: 00 213 41 29 94 72
E–mail: *ghalem1dz@gmail.com*

Esma Insaf Djebbar
Dept. of Computer Science, Faculty of Sciences,
University of Oran, Algeria
BP. 1524, El M'Naouer, Oran, Algeria (31000)
Phone: 00 213 41 29 94 72
E–mail: *d.insaf@yahoo.fr*

Abderahmann Benaissa
Dept. of Computer Science, Faculty of Sciences,
University of Oran, Algeria
BP. 1524, El M'Naouer, Oran, Algeria (31000)
Phone: 00 213 41 29 94 72
E–mail: *benaissa_inf@yahoo.fr*

Ali Cherif Brakeche
Dept. of Computer Science, Faculty of Sciences,
University of Oran, Algeria
BP. 1524, El M'Naouer, Oran, Algeria (31000)
Phone: 00 213 41 29 94 72
E–mail: *brakeche@gmail.com*