

Flooding Techniques for Resource Discovery on High Mobility MANETs

Rodolfo Oliveira, Luís Bernardo, Paulo Pinto
*Departamento de Engenharia Electrotécnica
Faculdade de Ciências e Tecnologia
Universidade Nova de Lisboa, Portugal
{rado,lflb,pfp}@uninova.pt*

Abstract

In this paper, we address high mobility AdHoc networks resource discovery problem. We use a virtual dynamic overlay network to perform broadcast flooding task optimization. Two new flooding techniques are proposed and compared with a global flooding strategy, and, a classical flooding scheme using source routing. Experimental results, obtained by several simulations implemented on ns-2, discourage the use of source routing in high mobility MANETs, and show that most of the times, flooding-optimization techniques could achieve best successful resource discovery rates than using global or source-routing flooding.

1. Introduction

Resource Discovery (RD) services are used to find the location of available resources on a network. If we consider mobile AdHoc networks (MANET) in general with their multi-hop wireless connections and absence of any fixed infrastructure, it can be difficult for the RD services to perform well. The problem gets worse if the nodes move fast towards each other and the system is considered a high mobility MANET (HMANET). To realize the importance of the RD performance, just consider disaster and emergency rescue operations, which are mission critical in nature. For example, in a forest fire, high mobility resources like fire-cars and aerial fire engines are spread on an area. RD services proposals for wired networks [1] are too static to be applied on a MANET, as they assume a priori defined links between the nodes.

Without a fixed network structure and having no knowledge about node's geographic position, resources are usually located flooding the network with a query message. Each node owning the searched resource

sends a hit message to the searcher node, usually following back the query's path (this technique is cited as source-routing). Source-routing decreases network loading, as it does not need to flood all network with hit messages. However, its lookup efficiency is too poor in HMANETs because link's changing frequency may damage the obtained reverse path. Another scenario consists on flooding all network nodes with the hit message (cited as pure flooding, because both query and hit messages are broadcasted by all nodes after receiving them). Apparently, it can increase lookup RD successful rate because flooded hit messages use all available paths. However, as dropped packets will increase with network load, pure flooding-based strategies can lead to lookup successful rates lower than expected. These two examples show the importance of node mobility and network load to the performance of RD services. It makes sense, then, to have different supporting algorithms to the RD services in HMANETs that adapt themselves to the specific conditions of the networks. Different flooding strategies must be dynamically switched depending on node's mobility behaviors and on network loading.

In this paper, we address HMANETs resource discovery problem using techniques implemented on MANET routing algorithms and some classical concepts used on peer-to-peer networks. We evaluate two new network-flooding strategies based on a virtual dynamic overlay network. Experimental results obtained by several simulations implemented on ns-2 [12] compare the proposed RD strategies with other two. The comparable aspects are: lookup efficiency, network loading, packet collision statistics and end-to-end delay.

The following section provides a related work survey, highlighting some HMANET's specific requirements that should be handled. Section 3 describes a set of flooding techniques evaluated on this paper. Section 4 describes some experimental scenarios

and evaluates the algorithm's performance. Finally, concluding remarks and future directions are given.

2. RD services requirements

RD is a recent research area in HMANETs. Most of research work made on HMANETs is related with network routing. Network routing can be viewed as a basic RD service: a target node must be located in a MANET before the source node asking for the route connects to it. Proactive routing schemes are proposed in DSDV [5] and CGSR [6] systems. DSR [7], ABR [8] and AODV [9] are reactive routing schemes, where new routes are created on demand using network flooding. Network knowledge is initially obtained using network flooding, and is cached to reduce the global flooding. Most of MANETs' routing protocols apply the source routing principles reversing the path obtained in the flooding phase. Therefore, they are vulnerable to node mobility. [10] compares DSDV [5], AODV [9] and DSR [7] routing protocols for various mobility metrics, and concludes that their routing performance squeezes with the increase on the relative node's velocity and frequency of link state changes on HMANETs.

RD systems are also used in peer-to-peer (P2P) network resource sharing systems. Some P2P works have been present last years to be applied in wired networks. Most P2P systems were designed for wired networks. Unstructured distributed systems like Gnutella [1] use global or partial flooding to perform resource location. Structured distributed systems, like CAN [17], and Pastry [18], avoid network flooding. They use distributed hash tables (DHTs) on each node, where routing information related with "*what node should be next visited in order to get a specific resource*" is contained. Structured systems have high maintenance costs to be applied on HMANETs, because link's changes will trigger table updates, which will consume too much bandwidth. Unstructured systems perform better in HMANETs because they do not use index services, avoiding index tables' maintenance costs.

Recently published works propose some P2P systems to work in ad hoc environments. Orion [3] presents an algorithm for construction and maintenance of an application-layer overlay network that enables routing of all types of messages required to operate a P2P system. Orion uses a combination of AODV routing protocol and IETF Simple Multicast and Broadcast protocol for MANETs. A node floods a query message with resources identification to locate, and on its way through the network the query message

sets up reverse paths to the query node. Orion is source routing based, and has low performance when applied in HMANETs, because reverse path can be damaged. Resource routing caching is used on each node (similar to routing tables used in AODV) but this is not suited to high mobility scenarios. On the other hand system scalability is improved, as application-layer is merged with network layer in order to reduce control overhead. KELOP [4] is a structured lookup system for MANETs that relies only on local information stored at each node to locate the closest estimates of the target. It reduces maintenance costs using information cached by the network layer routing protocols to route advertisements and lookups of resources based on DHT lazy update. With DHT reduced maintenance costs, KELOP system has low performance in HMANETs because routing caches get outdated very quickly, and most of routing protocols are based on source-routing. Finally, DPSR [2] is a routing algorithm for MANETs that uses P2P Pastry DHT system over DSR. The key idea of the integration is to bring the structured Pastry system to the network layer of MANETs, while retaining all the attributes of DSR for dealing with the specifics characteristics of MANETs. However, DSR is a pure source-routing protocol. For high mobility patterns, DSR presents low delivery packet ratio [10], which will increase DPSR network loading used for DHT maintenance.

It should be clear now that most of the RD services proposed for MANETs [2] [3] [4] are inefficient on HMANETs. There are important features that must be taken into account in the design of a RD system applied to HMANETs:

- Flooding should be used in a limited area. Flooding is not scalable in high diameter HMANETs since every message received at each node will be retransmitted, and gathering information from distant places can become useless quickly making the effort worthless. Unrestricted flooding is also a problem in MANETs, because it can origin a broadcast storm effect [15] that can reduce the performance for high traffic loads or in high node' densities scenarios;
- High mobility makes the source routing approach inefficient. Sometimes, the reverse path used in source routing schemes is not available because of the nodes' dynamics. For higher end-to-end delay values it gets worse. More reliable mechanisms are required to handle node unavailability;
- Routing cache procedures are dangerous. If they are refreshed at low frequency, they can present invalid data, conducting to erroneous routing decisions. Higher refreshing frequencies increase the network load without providing any reliability warranties.

Other redundant procedures must exist to increase system reliability;

3. RD flooding techniques overview

RD operation starts with a query message originated by a node. This message carries the node's address (n_{source}), the query identification (Q_{id}), and the resource identification to be located (R_{id}). Only nodes owning the required resource R_{id} will respond sending a hit message to inform the source node n_{source} about the resource's location. Four techniques are presented to route the originated query to the resource owner node (RD forward path), and the reply back to the source node (RD reverse path).

3.1. Classical approaches: Pure flooding and Flooding with source routing

Pure flooding is a brute force way to propagate queries and hits across the network. After receiving a query (or a hit message) nodes will blindly broadcast it. The query only carries n_{source} , Q_{id} , and R_{id} information. This technique does not use source routing. Every reachable node broadcasts all received queries and hit messages once. This technique is used to establish a benchmark level to evaluate the flooding optimizations proposed.

The flooding with source routing works this way: after receiving a query, a node will blindly broadcast it, as in pure flooding. During query propagation, each node puts its address in a query list containing all visited nodes. When a query reaches one resource owner node, the query list is copied to the hit message to set up the reverse path. The hit packet uses source routing based on the list.

3.2. Dynamic virtual overlay network for flooding optimization

The two new flooding techniques are based on a dynamic virtual overlay network (VON) and a short introduction is given in this section. The VON is based on the work described in [11], which proposes a soft-state clustering for MANETs using the concept of link stability proposed in ABR [8].

Network nodes automatically elect a node named Broadcast Group Leader (BGL), who manages a group of other nodes. A Broadcast Group (BG) is the set of nodes that elected the same BGL. Each node periodically broadcasts a short-beacon message. All nodes receiving a beacon from a node n_y 1-hop distant, are defined as being neighbor nodes of n_y . Nodes keep

a table of neighbors' link stability η . Link stability of neighbor node n_z is defined as the sum of consecutive beacons received from n_z . Link stability is set to null when more than one beacon is lost, producing a slight delay on link failure detection, but tolerating one beacon loss due to packet collision. A link is considered stable when η is above a `stability_threshold` and a node is stable when it has at least one stable link. In each beacon message a node sends its BGL address, and the higher link stability value contained in its table, which is represented by μ . Each node elects autonomously its BGL from the set of stable neighbor nodes. Before sending a new beacon, a node n_x elects its BGL applying the following rules:

- 1- When no BGLs exist in the neighborhood and stable links exist, n_x elects from the neighbor nodes that hold the highest value for η the one with the smallest address;
- 2- When no BGLs exist in the neighborhood and no stable links exist, n_x first compares if its neighbor's highest μ value is greater than `stability_threshold`. If it is, the lowest address neighbor node in the group with the highest μ value is elected. Otherwise, n_x will fail BGL election, and will not belong to any BG;
- 3- When node n_x is elected BGL by one of its neighbors and all its BGLs in the neighborhood have higher addresses than n_x , then n_x elects itself as its BGL (this is also applicable when no BGL exists in its neighborhood);
- 4- When rule 3 holds false and at least one BGL exists in the neighborhood, n_x elects the one with the smallest address;

The algorithm proposed sets BGs according to link stabilities (rule 1). When several BGs are overlapped, rule 4 will merge them, and observing rule 3, the BGL of the new merged BGs will be unique, as addresses are unique. This proposal originates less network load than classic clustering algorithms [16], because short beacon messages are used. As high link stability values represent low node mobility and vice-versa, the clustering proposed groups nodes according to their relative mobility, which is useful to perform flooding optimization in HMANETs. Left figure contained in figure 1 shows a HMANET where a VON was built. Dashed lines represent wireless links with neighbor nodes. Limited shadow areas represent BGs. There are three BGs. As it shows, BG overlapping can occur, since an elected BGL (nodes 5 and 6) can elect a lower address BGL (node 1) 1-hop distant. Node 9 has no BGL elected. Nodes 5, 1 and 6 are BGL nodes, and they will be used as a virtual overlay network.

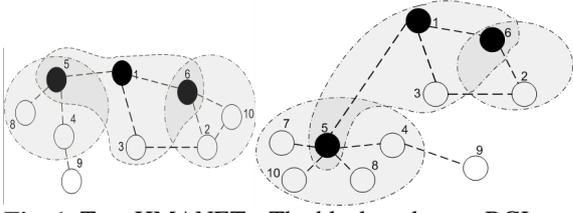


Fig. 1. Two HMANETs. The black nodes are BGLs.

3.3. 1-hop flooding optimization

Using the VON proposed in [11], it is possible to reduce the overall number of broadcasted queries. 1-hop flooding optimization task reduces the amount of broadcasts made by delaying some decisions. All BGL nodes will blindly broadcast the query. Non-BGL nodes only broadcast the query if during a waiting time period they do not receive enough information to avoid it. Suppose node 7, belonging to the right HMANET illustrated in figure 1, receives a query due a broadcast of BGL node 5. The query includes all BGL nodes visited, so, it contains the BGL node 5. After waiting time expiration, node 7 will avoid broadcasting because all its BGL neighbors (node 5) have been flooded. If node 3 receives a query originated on node 2, it will wait before take a decision. As delay is not applicable on BGL nodes, during the waiting time, node 3 has high probability of receiving the query from node 1. When the waiting period finishes, node 3 checks if all of its BGL neighbor nodes addresses (nodes 6 and 1) are contained in the queries received with the same identification Q_{id} and originated by the same n_{source} node. If so, broadcast will be avoided, otherwise it will broadcast the query.

To implement this technique, queries must contain the previously described identifiers n_{source} , Q_{id} , and R_{id} plus two distinct lists (fifo queues). List S contains all nodes BGLs visited by the query. List U holds all non-BGL nodes visited since last BGL node. BGL nodes will simply remove and cache list U contained in the query, and after adding its address to S list they blindly-broadcast the query. BGL and non-BGL nodes also hold the last broadcasts Q_{id} originated in n_{source} node to just broadcast the query once. When each non-BGL receives a new query it temporarily holds its identifiers and its lists (S_{cache}, U_{cache}). To reduce query broadcasts, non-BGLs nodes delay query-broadcasting procedure during a t_{delay} value. During t_{delay} a non-BGL will overhear its neighbor broadcast queries with the same n_{source} and Q_{id} and it will analyze if all nodes contained in S list of received queries are contained in its initially cached S_{cache} list. Each non-BGL node builds a list K when a new query is received. K list will hold all nodes contained in lists S available on future

repeated receiving queries, which are not included in the list S_{cache} . Lists S_{cache} and K cached in non-BGLs contain the knowledge about BGLs visited by that query. When delay expires, a non-BGL determines if the query has been already broadcasted by each of its neighbors' BGL. The non-BGL node will broadcast the query if at least one of its neighbors' BGL did not broadcast that query. Query will be broadcasted if exists one element of BGL neighbors list (built with the information contained in beacon messages) that is not contained in lists S_{cache} or K .

When a node owning the requested resource receives the query, it will use lists U and S to propagate a hit message using source routing. In each BGL, the list previously cached is restored and added to hit message. To propagate hit messages, a node uses a unicast acknowledge scheme. When hit message are not acknowledged by its receiving neighbor node, a node blindly broadcasts the hit to find an alternative path. Original reverse path may be restored, if after being blindly broadcasted a hit message reaches one of nodes contained in both lists S or U .

3.4. 2.5-hops flooding optimization

To have neighborhood information, each node sends 2 hops BGL neighborhood information on each beacon message used to build the OVN. Each beacon message sent includes all BGLs chosen by its neighbor nodes. For example, node 1 illustrated in right HMANET of figure 1 will broadcast its beacon containing BGLs 6, 5 and 1. A node not included in any BG, as node 9, is considered a BG of just one node.

BGLs always broadcast all received queries. Non-BGL nodes only broadcast a received query if it contains an order to do it. The query's sender uses the available BGLs knowledge 2-hops away to select who should broadcast the query. Suppose that node 5 of right HMANET presented in figure 1 origins a query to propagate over all nodes. Node 5 has knowledge about BGLs 1, 6 and 9, and also knows that only node 4 can flood BGL node 9, because in the beacon received by node 5, node 4 has previously announced about its BGL neighbor 9. As node 4 is not a BGL, node 5 will include it in the list of nodes that should broadcast the query. The algorithm is presented in figure 2 and is computed when node origins a new query, a BGL broadcasts a query, or when a node receiving a query is appointed as a broadcaster. It returns list v , with the neighbor non-BGLs that must broadcast the query.

To propagate the hit message, this technique follows the same algorithm described in 1-hop optimization.

```

1. build_list_of_BGLs_2hops_away(n)
2. erase_list(n,S)//erase all BGL visited
3. for each_lhop_neighbor h_x
4.   remove_elem(n,h_x)//erase all neighbor
5. remove_elem(n,n_a)//n_a can be auto-elected
6. for each_BGL_contained_in_list_n bgl_x
7.   for each_lhop_neighbor h_x
8.     test_if_bgl_x_is_only_reachable_from_h_x
9.     if bgl_x_is_only_reachable_from_h_x
10.      add_elem(v,h_x)
11.      remove_elem(n, h_x)
12. while not_empty(n)
13.   ng = next_to_reach_more_BGL(n)
14.   add_elem(v,ng)
15.   list_reached = compute_BGL_reached_by(ng)
16.   erase_list(n,list_reached)

```

Fig. 2. Algorithm applied in node n_a to select neighbors that must broadcast the query.

4 Performance Evaluation

Several simulations were made using *ns-2* platform. In each simulation scenario, 200 nodes are moving during 1000 seconds on a 1000m x 1000m area according to the Generalized Random Waypoint mobility model [14]. Four different mobility scenarios were defined to study the mobility behavior of each flooding technique. Node's average speeds of 0m/s, 5 m/s, 20m/s and 50m/s were obtained using constant pause times of 5 seconds. Each node has approximately 100 meters of communication range using IEEE 802.11b and the *two-ray ground* propagation model.

VON infrastructure implemented in *ns-2* uses a beaconing frequency of 1 Hz. Ten thousands of different resources are randomly distributed on the network nodes. Three different behavior patterns were defined using the model presented in [13], which proposes a uniformly way to generate queries in each node. 10927, 1125 and 267 generated queries are used to perform simulations with high, medium and low network loading, respectively. Finally, all broadcasts are sent with a variable jitter bounded to 100 ms. 1-hop flooding technique uses a constant t_{delay} value of 700 ms.

Figure 3 presents experimental results. Graphics a, b and c show the RD successful rate for each flooding technique and using different loads. We can observe that source routing decreases its performance for high mobility patterns. 2.5-hop optimization also presents low successful rates for high mobility values. For average speed of 50 m/s, 2.5-hop technique has low performance, because it uses outdated VON information to decide who should broadcast the query, since beaconing frequency is too low for that mobility values. Pure flooding is the most redundant technique and presents the best successful rates. 1-hop

optimization technique has successful values sometimes greater than pure flooding, but for higher network loads its performance decreases.

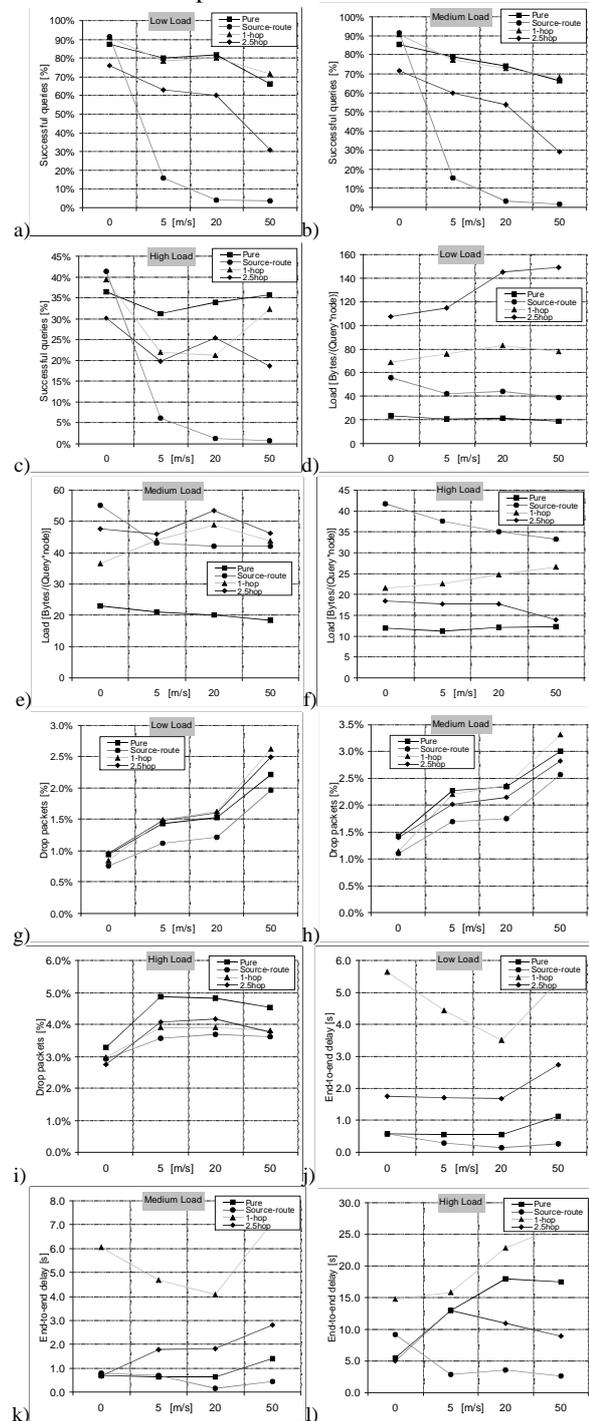


Fig. 3. Experimental results for low, medium and high network loads; Successful rate (a, b, c); Network load per query-node (d, e, f); Packet loss (g, h, i); End-to-end delay (j, k, l);

VON performance is reduced using higher loads, since packet collisions (graphic i) increases, which degrades 1-hop and 2.5-hop performance. Pure flooding performance also decreases using high loads since it presents the higher rate of dropped packets by collisions. The backoff timer window used in IEEE 802.11 increases when collisions are detected, which also increases the end-to-end delay measured in all flooding techniques (graphic l).

Graphics d, e and f show the loading cost per query-node, including the VON maintenance. For medium and high loads, the cost of source routing is similar to the cost of 1-hop and 2.5-hop techniques, even not using a VON. For low loads, techniques using the VON are penalized because the cost to maintain the VON is too high when few queries are originated. Pure flooding has the lowest costs. The use of short messages, and the needlessness of a VON infrastructure decreases the cost, even when pure flooding is the technique where more query broadcasts are made.

5. Conclusions and further work

The results presented in this paper have significant relevance for RD in HMANETs. First, they discourage the use of source routing to perform RD in high mobility MANETs. Second, pure flooding performs comparatively well and presents lower costs. Pure flooding is used as a benchmark and is not appointed as a solution, because although it achieves good results, it does not give enough features to use in an optimization process.

This paper proposes 2 new flooding strategies based on a VON. 1-hop is an adaptive algorithm, which beats source routing for low mobility nodes due to clustering use; and that behaves like pure flooding for high mobility nodes, reacting to VON efficiency decay and answer path routing failure. On the other hand, 2.5-hop also reacts to VON efficiency decay and answer path routing failure, but presents lower successful rates and is less redundant.

In the proposed techniques, VON has not been completely explored. Further work is also being done developing RD algorithms where a DHT approach may be used on top of a stable virtual overlay network.

References

[1] T. Klingber, R. Manfredi, "RFC Draft of Gnutella v0.6". http://rfc-gnutella.sourceforge.net/src/rfc-0_6-draft.html, 2002.

- [2] A. Klemm, C. Lindemann, O. Waldhorst, "A Special-Purpose Peer-to-Peer File Sharing System for Mobile Ad Hoc Networks", *Proc. Workshop on Mobile Ad Hoc Networking and Computing*, France, 2003.
- [3] Y. Charlie, S. Das. H. Pucha, "Exploiting the synergy between Peer-to-Peer and Mobile Ad Hoc Networks", *Proc. of HotOS-IX*, Hawaii, 2003.
- [4] S. Bashir, B. Li: KELOP, "Distributed Key-Value Lookup in Wireless Ad Hoc Networks", *Proc. of the 12th IEEE Int. Conf. on Computer Communications and Networks*, Texas, 2003.
- [5] P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance Vector Routing (DSDV) for Mobile Computers", *Proc. of ACM SIGCOMM*, pp. 234-244, 1994.
- [6] C.-C. Chiang, H.-K. Hu, W. Liu, M. Gerla, "Routing in Clustered Multihop Mobile Wireless Networks with Fading Channel", *Proc. of IEEE Singapore Int. Conf. on Networks*, 1997.
- [7] D. Johnson, D. Maltz. Mobile Computing, "Kluwer academic publishers", 1996.
- [8] C.-K. Toh, "Associativity-Based Routing for Ad-hoc Mobile Networks", *Journal of Wireless Personal Communications*, vol. 4, 1997.
- [9] C. Perkins, E. Royer, "Ad-Hoc On-Demand Distance Vector Routing", *Proc. of 2nd IEEE Workshop on Mobile Computing Systems and Applications*, 1999.
- [10] J. Tsumochiy, K. Masayama, H. Uehara, M. Yokoyama, "Impact of Mobility Metric on Routing Protocols for Mobile Ad Hoc Networks", *Proc. IEEE Pacific Rim Conf. on Communications, Computers and Signal Processing*, Victoria, 2003.
11. L. Bernardo, R. Oliveira: Technical Report, http://tele1.dee.fct.unl.pt/tg_group/html/papers.html#Tech-reports/TR-2005-01.pdf. NOTE TO THE REVIEWERS: The contents of this technical report were submitted to ICETE'2005, and if accepted, this reference will be replaced on the camera-ready version.
- [12] "The Network Simulator ns-s", <http://www.isi.edu/nsnam/ns>, 2005.
- [13] Z. Ge, D. Figueiredo, S. Jaiswal, J. Kurose, B. Towsley, "Modeling Peer-Peer File Systems", *Proc of IEEE INFOCOM*, San Francisco, March-April, 2003.
- [14] J. Yoon, M. Liu, B. Noble, "Random Waypoint Considered Harmful", *Proc. of IEEE INFOCOM*, San Francisco, March-April, 2003.
- [15] Tseng, Y.-C., Ni, S.-Y., Chen, Y.-S., Sheu, J.-P., "The Broadcast Storm Problem in a Mobile Ad Hoc Network", in *Wireless Networks*, vol. 8, pp. 153-167, Mars-May 2002.
- [16] Wu, J., and Lou, W., "Forward-node-set-based broadcast in clustered mobile ad hoc networks", *Wireless Communications and Mobile Computing*, N.3, pp.155-173, 2003.
- [17] S. Ratnasamy, P. Francis, M. Handley, R. Karp, S. Shenker, "A scalable content-addressable network", *Proc. of ACM SIGCOMM*, 2001.
- [18] A. Rowstron, P. Druschel, "Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems", *IFIP/ACM Int. Conf. on Distributed Systems Platforms*, Heidelberg, Germany, November, 2001.