We are IntechOpen, the world's leading publisher of Open Access books
Built by scientists, for scientists

13

An Adaptive Broadcasting Scheme in Mobile Ad Hoc Networks

Dimitrios Liarokapis, and Ali Shahrabi
Glasgow Caledonian University
United Kingdom

1. Introduction

Mobile and static nodes in battlefields or within the vicinity of disaster areas may not depend on fixed infrastructure for communication. To rapidly provide the required communication between the nodes in such environments, a Mobile Ad hoc Network (MANET) is the only available platform. With no fixed infrastructure, the efficient use of MANETs resources is highly crucial for the successful communication between mobile nodes. In situations where both the transmitting and the receiving nodes are placed within the transmission range of each other, communication is possible through a single-hop connection. In all other scenarios where the nodes are distanced, the exchange of packets is possible as long as a multi-hop path is available between them. Despite the unique characteristics of MANETs, they share many attributes and operations with other traditional networks. DNS lookups, exchange of control packets for management purposes and routing discovery requests are some examples of common operations, which all require broadcasting pieces of information across the network. However, due to lack of a centralised administrative and hardware, some modification is required to adopt broadcast operation for MANET environment.

The most straightforward broadcast mechanism used in MANETs is Simple Flooding (SF). The algorithmic procedure followed in SF is very simple, thus making its implementation and integration inside more complex operations fairly undiscomforting. In SF, upon reception of a broadcast packet the receiver will check whether or not this is a duplicate packet. If it is a new packet it will immediately retransmit it to all of its neighbouring nodes. Simply flooding the entire network may be the fastest and easiest way for a node to broadcast information over the network but it has been found to be a very unreliable and resource inefficient mechanism leading to the Broadcast Storm Problem (Ni et al., 1999) especially in highly populated and dense networks.

Over the past few years many studies (Leng et al., 2004), (Zhu et al., 2004), (Qayyum et al., 2002), (Hsu et al., 2005), (Purtoosi et al., 2006), (Barrit et al., 2006), (Bauer et al., 2005) have proposed novel broadcast mechanisms to alleviate the effects of SF. Early works were focused on developing schemes where the rebroadcast decision is made based on fixed and pre-determined threshold values. Probability-Based (PB), Counter-Based (CB) and Distance-Based (DB) are three schemes which have been proposed based on the concept of introducing a threshold value. PB bases it rebroadcasting decision on a fixed probability value, CB decides it by counting the number of received duplicate packets and finally the rebroadcasting in DB is based on the distance between sender and receiver (Ni et al., 1999). All of these schemes were found to considerably improve the performance of the broadcast operation in various network topologies but they also introduced a new dependency. The threshold value to be selected in order to reach optimum overall network performance highly depends on traffic load volume and node population. The degree of dependency is such that in certain network topologies SF performs better than these schemes (Williams et al., 2002).

The development of threshold-based adaptive broadcast schemes has consequently been considered to alleviate these dependencies. According to their algorithmic procedures, these schemes adaptively adjust the threshold value to be used depending on local information with regards to the density of the network within the transmission range of the sender (number of one-hop neighbours) or within a broader network area (number of two- hop neighbours) or even within the entire topology. Hence, all the schemes can be categorized based on the mechanism used in order to implement adaptivity. The most commonly used mechanism implies all nodes to periodically exchange HELLO packets with their neighbouring nodes in order to calculate density (Ryu et al., 2004), (Lee et al., 2006), (Chen et al., 2002), (Colagrosso 2007), (Chen et al., 2003), (Kyasanur et al., 2006), (Tseng et al., 2003). Alternatively, the other group of adaptive broadcast schemes utilise a positioning system, e.g. GPS, resulting in the construction of a network map for every mobile node, calculating in that way a very precise value for the density of the network (Deng et al., 2006). These schemes either introduce more overhead traffic to the network or demand the existence of expensive and fairly unreliable positioning systems.

In this chapter, a novel Distance-Based Adaptive (DibA) scheme is proposed. Based on the Distance-Based broadcast scheme, DibA implements adaptivity by dynamically adjusting the distance threshold value for every rebroadcast operation independently. Knowledge on local network densities is created on demand, without relying on HELLO packets or GPS systems, thus making DibA highly reliable avoiding at the same time the introduction of extra overhead traffic.

The remainder of this chapter is organised as follows. In Section 2 we overview related work. DibA as an adaptive broadcast mechanism is introduced in Section 3. In Section 4 the process of building a highly diverse network topology, where the performance of adaptive schemes can be evaluated appropriately is explained. The performance study is presented in Section 5. Finally, we make concluding remarks in Section 6.

2. Related works

In this section the Distance-Based scheme will be presented in detail, as our proposed scheme enhances this algorithm in order to make it locally adaptive. We will also discuss the general characteristics of other adaptive schemes and their methods.

2.1 Distance-Based scheme

DB is a broadcast mechanism that uses the distance between sender and receiver to make the decision whether to rebroadcast or not (Ni et al., 1999). The power of the received signal is a parameter that can be used to calculate the distance. GPS can also be used for that purpose. The specific algorithm for DB is presented in Fig. 1.

Algorithm: DB
Input: broadcast message (*msg*)
Output: decides whether to rebroadcast or not

S1. When a broadcast message, *msg*, is heard for the first time, initialize d_{min} to the distance of the broadcasting node. If $d_{min} < D$ (where D is the distance threshold), proceed to S5. In S2, if *msg* is heard again, interrupt the waiting and perform S4.
S2. Wait for a random number of slots. Then submit *msg* for transmission and wait until the transmission actually starts.
S3. The message is on the air. The procedure exits.
S4. Update d_{min} if the distance to the host from which *msg* is heard is smaller. If $d_{min} < D$, proceed to S5. Otherwise, resume the waiting in S2.
S5. Cancel the transmission of *msg* if it was submitted in S2. The host is inhibited from rebroadcasting the message. Then exit.

Fig. 1. DB Algorithm

The distance threshold used in DB, is a parameter valued by default. This value is fixed and does not change unless there is administrative intervention. This is the major drawback of DB, as a static threshold may be appropriate for a network of specific density under particular circumstances. It could potentially cause poor network performance when the density or other network conditions greatly differ (Williams et al., 2002).

2.2 Adaptive schemes

Over the past few years, a growing number of studies have been trying to develop adaptive versions of DB. In order to achieve this, having the instantaneous knowledge of network configuration (in particular, the number of mobile nodes placed within the transmission range of each sender) is required. Currently, there are only two methods used to determine the local density for every individual node.
The first mechanism makes use of a positioning system such as Global Positioning Systems (GPS) (Deng et al., 2006). Mobile nodes periodically exchange messages including their exact coordinates. When a mobile node receives these coordinates it can calculate the distance from its current position and decides if the transmitting node is placed inside the transmission radius. In case that is true, the node increases its neighbours counter and therefore it can determine the level of network density locally. The use of expensive positioning systems, such as GPS, is the limitation of this approach.
According to the second mechanism (Ryu et al., 2004), (lee et al., 2006), (Chen et al., 2002), (Colagrosso 2007), (Chen et al., 2003), (Kyasanur et al., 2006), (tseng et al., 2003) the mobile nodes need to periodically send HELLO packets to all their neighbouring nodes and consequently count the number of responses they receive to measure the local density. It is obvious that this approach introduces a significant amount of overhead traffic in the network that could negatively affect the overall network performance, especially in cases where the network is highly populated and already overwhelmed with other types of traffic. In addition, one also needs to decide on the frequency of this procedure to take place. It should be remembered that although an increase in performance is the net result of introducing overhead (i.e. HELLO packets) and reducing overhead (i.e. fewer rebroadcasting), a frequent transmission of HELLO packets in static networks only increases the amount of overhead.

Although both supporting mechanisms exploit adaptivity, they also have significant drawbacks that could produce additional constraints. In the next section, we propose a novel broadcast algorithm which is neither relying on any positioning system nor introducing overhead traffic.

3. Distance-based Adaptive scheme (DibA)

To perform adaptively without introducing any further constraints and in order to decide whether or not to rebroadcast a message, any broadcast scheme requires to provide information about the local density of network for every node.

In our approach, we make use of Step 2 (S2) of the DB original algorithm, presented in Fig. 1, and make minor changes to Step 4 (S4). According to DB in S2, the receiving mobile node needs to wait for a random number of slots and remains in listening mode for duplicate broadcast packets. During that period of time, upon reception of a duplicate packet, it calculates the new distance and compares it with the distance threshold *D*.

We take advantage of this waiting period and calculate the number of duplicate packets received, using a simple counter which is updated in S4. The number of identical packets arriving at the mobile node is closely connected to the number of neighbouring nodes. Each time the value of the counter increases, the distance threshold is tuned according to a specific pattern.

The increase or decrease of the distance threshold is closely related to the potential additional coverage area that could be achieved when the broadcast packet is transmitted. If a large extra area is predicted to be covered by rebroadcasting of a packet, the distance threshold should be set to a low value. That is the case when the counter value is low. On the contrary, if the predicted coverage area is small, the distance threshold should be adjusted to a high value. This is also the case when the counter value is high. It is obvious that counter value, distance threshold and extra coverage area greatly affect one another in that order.

The DibA algorithm makes use of a scaled if statement for the adjustment of the distance. This should lead to an exponential increase of the distance threshold depending on the counter value. An example of the scaled if statement is as follows.

```
if(count = 1)
        D = 50m;
else if(count < 4)
        D = 125m;
else
        D =200m;
```

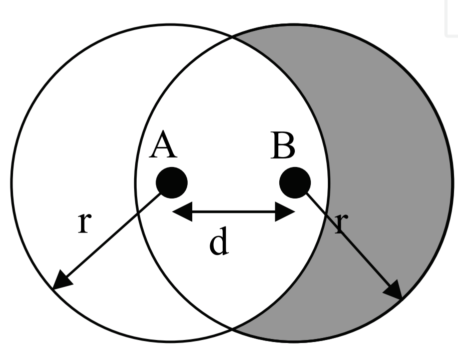


Fig. 2. Extra Coverage Area Analysis

In order to justify the reason why this pattern is used, we need to take into consideration the redundant rebroadcast analysis performed in (Ni et al., 1999). Consider the scenario in Fig. 2. Node A sends a broadcast packet and node B decides to rebroadcast it. Let S_A and S_B denote the circle areas covered by the transmission ranges of nodes A and B respectively. The gray area represents the additional area that will be covered by B's rebroadcast named S_{B-A}. We can derive that:

$$|S_{B-A}| = \pi r^2 - INTC(d)$$

where *INCT(d)* is the intersection area of the two circles centred at two points distanced by d.

$$INTC(d) = 4\int_{d/2}^{r} \sqrt{r^2 - x^2}\,dx$$

The extra coverage area gets the maximum value when r = d and is equal to:

$$\pi r^2 - INTC(r) = r^2\left(\frac{\pi}{3} + \frac{\sqrt{3}}{2}\right) \approx 0.61\pi r^2$$

Thus, B's rebroadcast can cover an extra area of 61% of the area covered by the previous transmission. The average extra coverage area can be obtained by integrating the above value over the circle of radius *x* centred at A for *x* in [0, r]:

$$\int_0^r \frac{2\pi x \cdot \left[\pi r^2 - INTC(x)\right]}{\pi r^2}\,dx \approx 0.41\pi r^2$$

A rebroadcast can cover an additional of 41% area in average. Following the same pattern, the extra area covered can be calculated depending on the number of transmissions heard for the broadcast packet. The result is shown in the graph of Fig. 3.

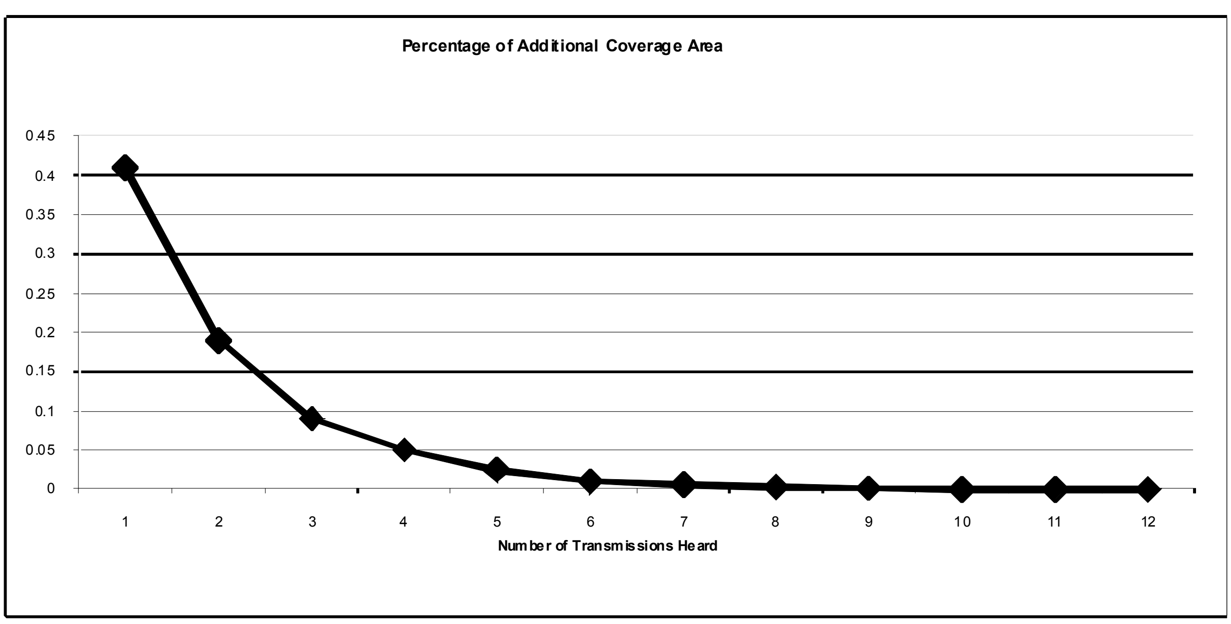


Fig. 3. Analysis of Redundant Rebroadcasts

Algorithm: DibA
Input: broadcast message (*msg*)
Output: decides whether to rebroadcast *msg* or not

S1. When a broadcast message *msg* is heard for the first time, initialize d_{min} to the distance of the broadcasting node and the count to 1. If $d_{min} < D$ (where D is the distance threshold), proceed to S5. In S2, if *msg* is heard again, interrupt the waiting, increase *count* by 1 and perform S4.
S2. Wait for a random number of slots. Then submit *msg* for transmission and wait until the transmission actually starts.
S3. The message is on the air. The procedure exits.
S4. Update d_{min} if the distance to the host from which *msg* is heard is smaller.
If *count* is less than c_1
then $D \leftarrow D_1$
else if *count* is less than c_2
then $D \leftarrow D_2$
else …
……..
else if *count* is greater than c_n
then $D \leftarrow D_n$
If $d_{min} < D$, proceed to S5. Otherwise, resume the waiting in S2.
S5. Cancel the transmission of *msg* if it was submitted in S2. The host is inhibited from rebroadcasting message. Then exit.

Fig. 4. DibA Algorithm

The value of the distance threshold could change multiple times during the waiting period and every time a duplicate broadcast packet is received, the distance between sender and receiver is compared with the current value of the threshold. The details of DibA algorithm are presented in Fig. 4, where D is the distance threshold, *count* is the counter described above, D_1, D_2 … D_n are the predetermined threshold values and c_1, c_2 … c_n are predetermined counter values.

DibA's primary goal is not to calculate accurately the number of neighbouring nodes, but to decide upon the density level of the network locally inside the transmission radius. This feature gives an extra advantage to our approach in comparison to other adaptive schemes.

Let us consider part of a network topology as shown in Fig. 5. This is an extremely diverse topology as in the right part of the network only 1 node is placed. The left part of the network covers 12 nodes. All nodes have the same transmission range TR. The black node (BN) sends a broadcast message that will be received by all of its neighbouring nodes. In this example, the only neighbour of BN is the grey node (GN).

When we use one of the already existing adaptive schemes, GN will try to calculate the exact number of nodes inside the transmission radius. Either using GPS or HELLO packets, the end result of the calculation will be very close to 12, the total of all white nodes (WN) and BN. As a result, GN will decide that the network is very dense locally and tune the distance threshold to be high, in order to rebroadcast only if it is placed at the edge of BN's transmission range. In case that the distance between BN and GN is not large enough to exceed the tuned distance threshold (Fig. 6), GN will not rebroadcast. None of the WNs will receive the broadcast packet.

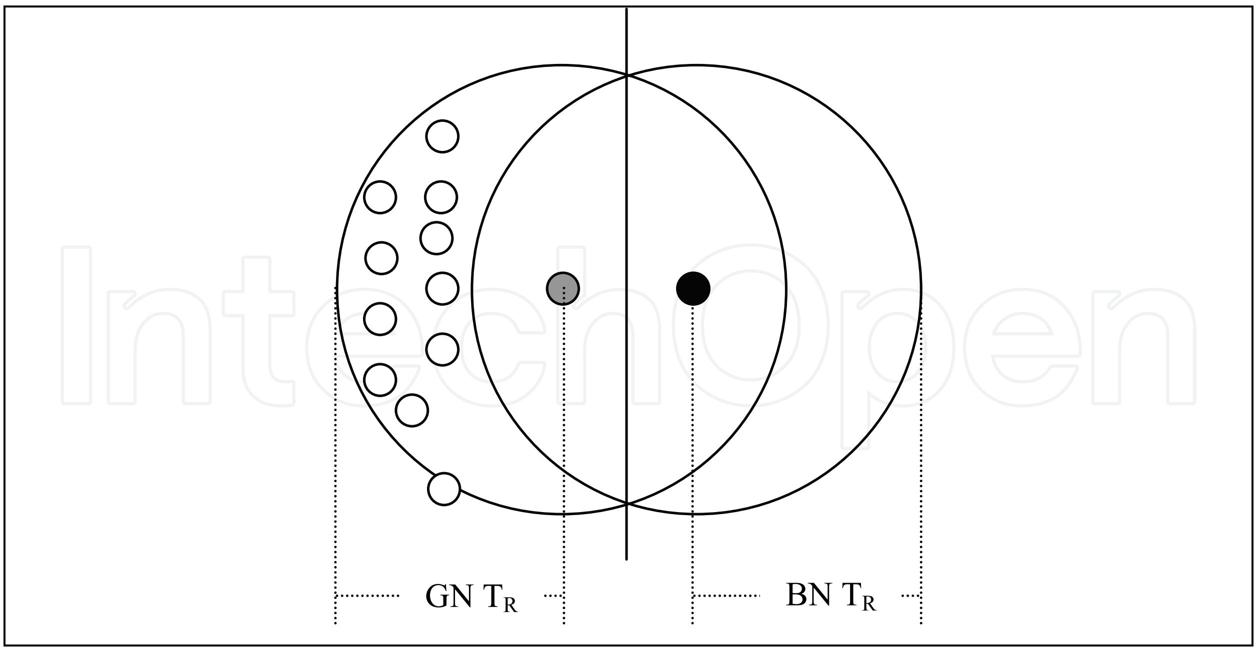


Fig. 5. Diverse Network Topology

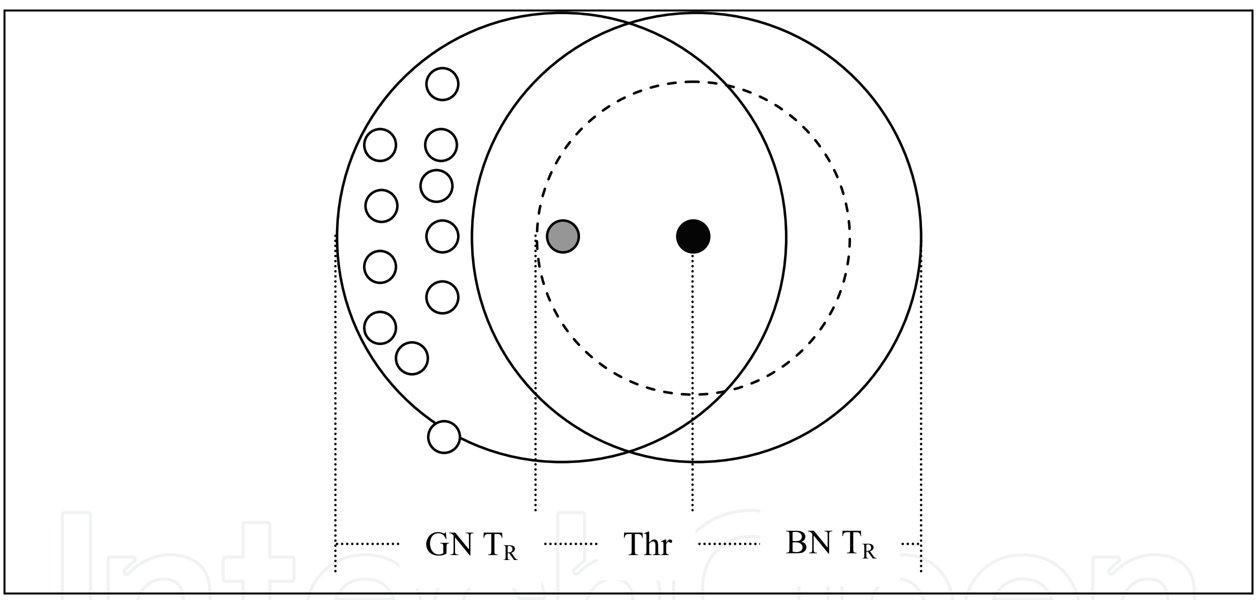


Fig. 6. Existing Adaptive Schemes

In case that DibA is used as the broadcast scheme, after reception, GN will wait for a random period of time counting duplicate packets. As BN is the only neighbour that has broadcasted the packet, GN exits the listening mode with the counter value of 1. It then assigns a very low value for the distance threshold. Now, it is highly possible at this point, as the threshold is very low, that GN is placed outside the dotted circle, as shown in Fig. 7. As a result, GN will rebroadcast the packet and all WNs will receive it.

In this example, we have shown that knowing the exact number of neighbouring nodes is not always ideal when trying to decide upon the appropriate value for the distance threshold. DibA measures the level of local density, depending on duplicate receptions and not on the knowledge about the amount of neighbours. Thus, it is highly reliable for both normal and extremely diverse network topologies.

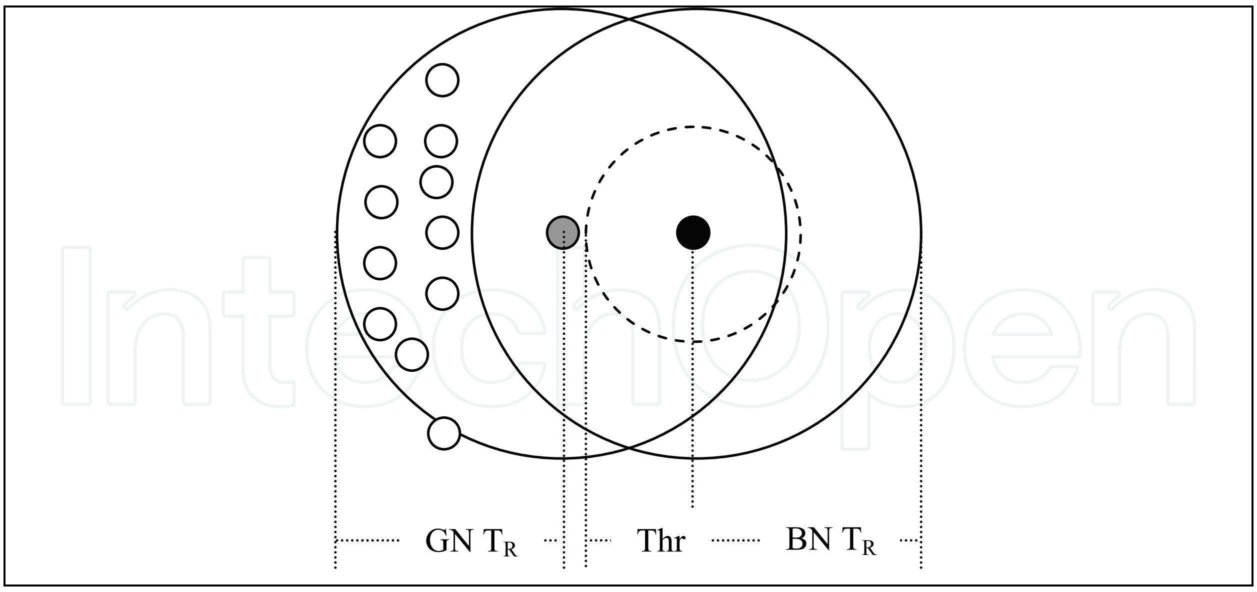


Fig. 7. DibA

4. Building a diverse network topology

Most of studies (Ni et al., 1999), (Leng et al., 2004), (Zhu et al., 2004), (Qayyum et al., 2002), (Hsu et al., 2005), (Purtoosi et al., 2006), (Barrit et al., 2006), (Ryu et al., 2004), (Lee et al., 2006), (Chen et al., 2002), (Colagrosso 2007), (Chen et al., 2003), (kyasanur et al., 2006), (Tseng et al., 2003) are relying on a simple network topology consisted of nodes distributed nearly evenly in an area when studying the performance of a broadcast scheme. However, the performance of any adaptive scheme is more appropriately demonstrated when tested on a diverse network topology, where part or parts of the network significantly differ in mobile nodes population volumes. In this section, we present the implementation of an automatic mechanism that can be used to create this kind of topologies.

The simulation tool that we use for our experiments is NS-2.30. NS-2 offers a single tool for creating mobility files using the setdest command. The user has the options to select the length and width of the topology, the number of nodes, pause time, maximum and minimum speed and simulation time. Unfortunately, setdest does not provide options to create more complex scenarios. However, the mobility files generated are of a simple text format, which gives us the opportunity to manually intervene inside the files and make appropriate changes.

The structure of the mobility file is as follows. Every node is assigned with its initial X, Y, Z coordinates in a command line. For example:

at 0.0 (time) node(0) 2.345 4.123 0.0

After all nodes are assigned initial coordinates, setdest randomly selects the time point where each node will change its direction and speed in order to reach a specific (X, Y, Z) point inside the topology. An example of such a command line is:

at 3.4567 (time) node (0) 4.899 13.756 10.392

Where the first parameter after “node(0)” (4.899) is the X coordinate for the reaching point, the second parameter (13.756) is the Y coordinate for the reaching point and the third

parameter (10.392) is the speed of the mobile node. We have not included other parameters that are of no significance for the movement of the nodes in our examples.
We will explain how our mechanism works using a simple example. Let us consider the case where we want to create the topology presented in Fig. 8.
The nodes need to move inside their own half of the network. The fact that there is limitation of movement using borders helps to keep a balanced percentage of

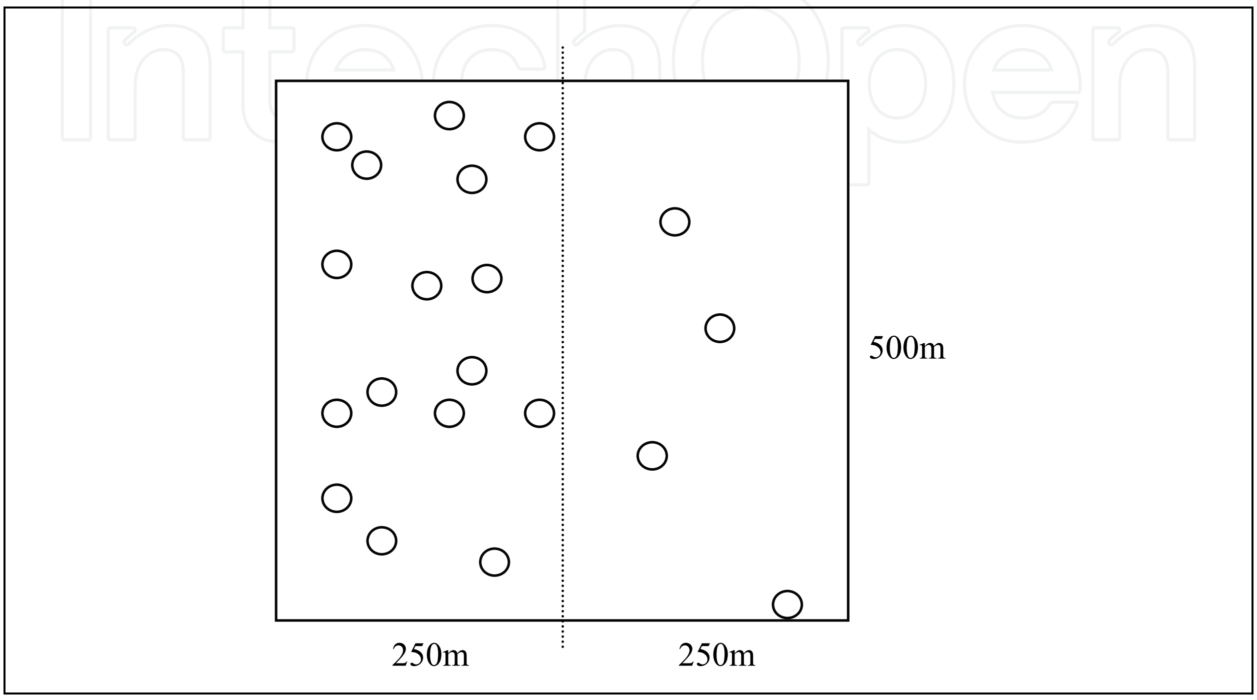


Fig. 8. A Sample Diverse Topology

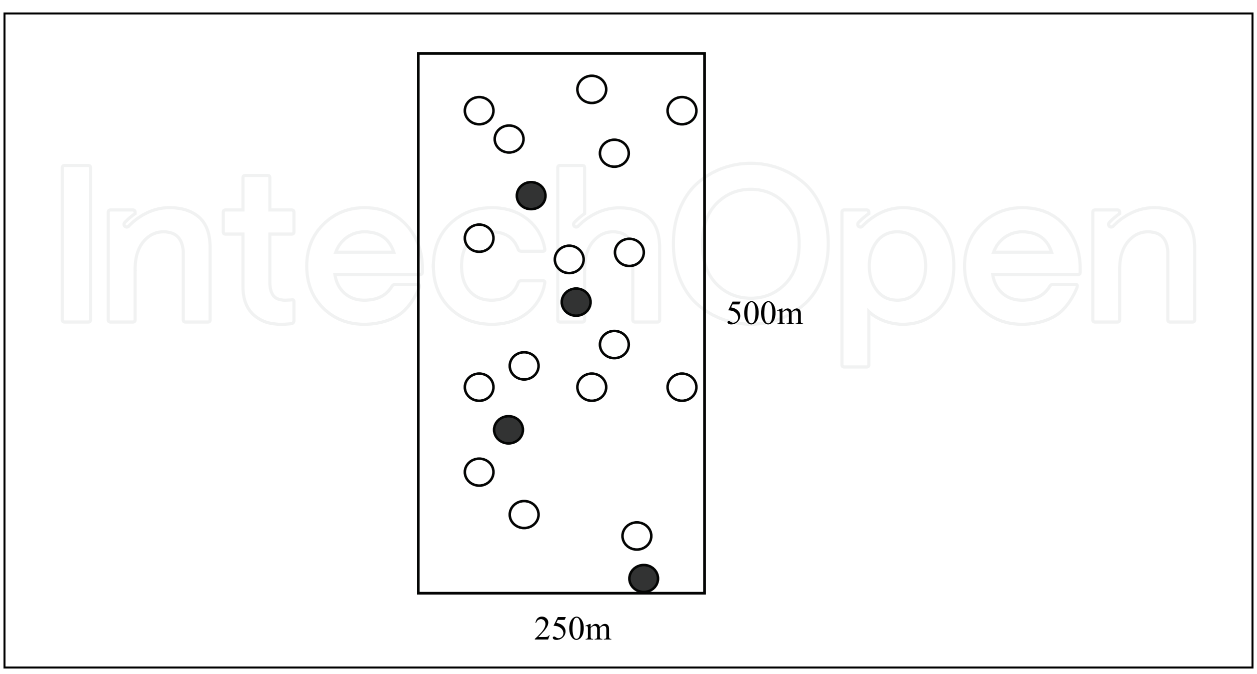


Fig. 9. Base Topology

differentiation. Simulation results are not affected, as the traffic generated is not unicast or multicast but broadcast. Our main goal is for all the mobile nodes to receive the broadcast packet.
In the above topology, 20% of the mobile nodes (4/20) are placed inside the right part of the network and 80% of them (16/20) are placed inside the left part. In order to create this topology, we need to start from a base topology as presented in Fig. 9.
The volume of diversity is then specified by selecting an appropriate percentage of the mobile nodes, which in our example is 20%. These are the black nodes of Fig. 9. We developed a simple software tool that scans the mobility file for all the command lines that either initialize or change the movement of all 4 black nodes. The value of X in these command lines is then increased by 250m, in order to migrate the black nodes over to a topology of identical length and width that touches the base topology vertically. Fig. 10 shows the migration process.
The movement of the black nodes initially is limited with regards to the X coordinate between 0m and 250m. Thus, after the modification of the mobility file, these nodes are restrained to move inside the right half of the topology, with the X value varying between 250m and 500m.
As a result of the process described above we get the end result of Fig. 8.

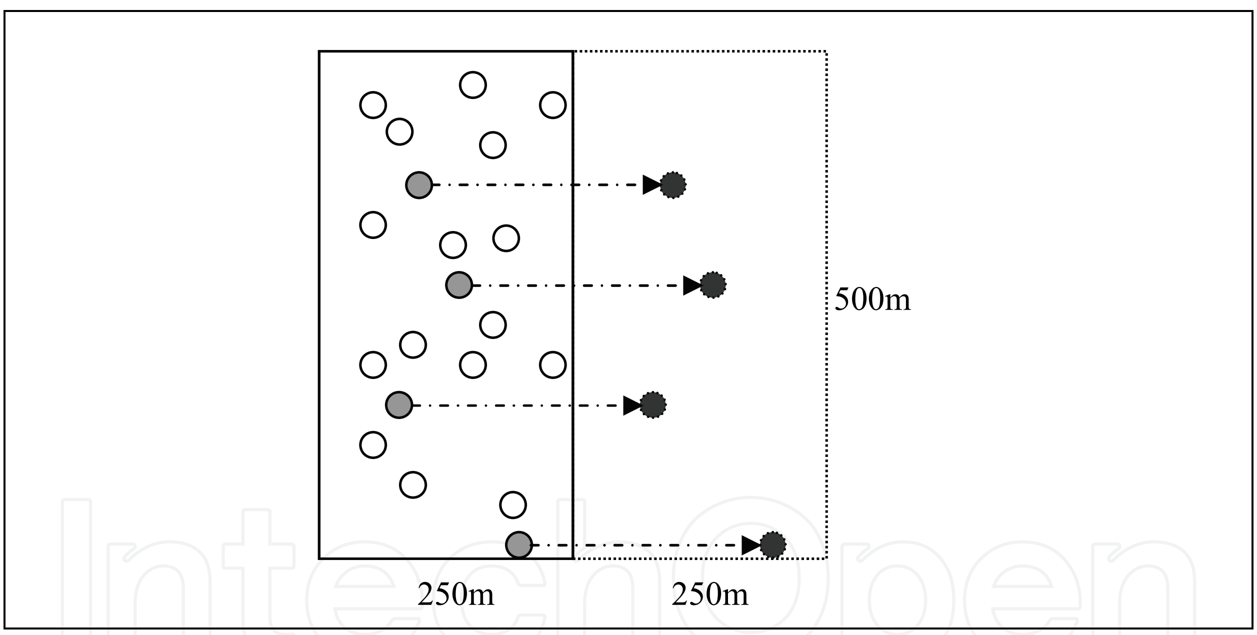


Fig. 10. Migration Process

5. Performance analysis

We implemented Distance-Based Adaptive scheme (DibA) and Distance-Based scheme (DB) using the network simulator NS2.30. We have used the NS2 code for DB provided by (Barrit et al., 2006), (Williams et al., 2002).

5.1 Simulation set-up and parameters

Node mobility is simulated using mobility files that are generated by the NS2 mobility generation feature setdest. Our experiments make use of both normal and diverse topologies,

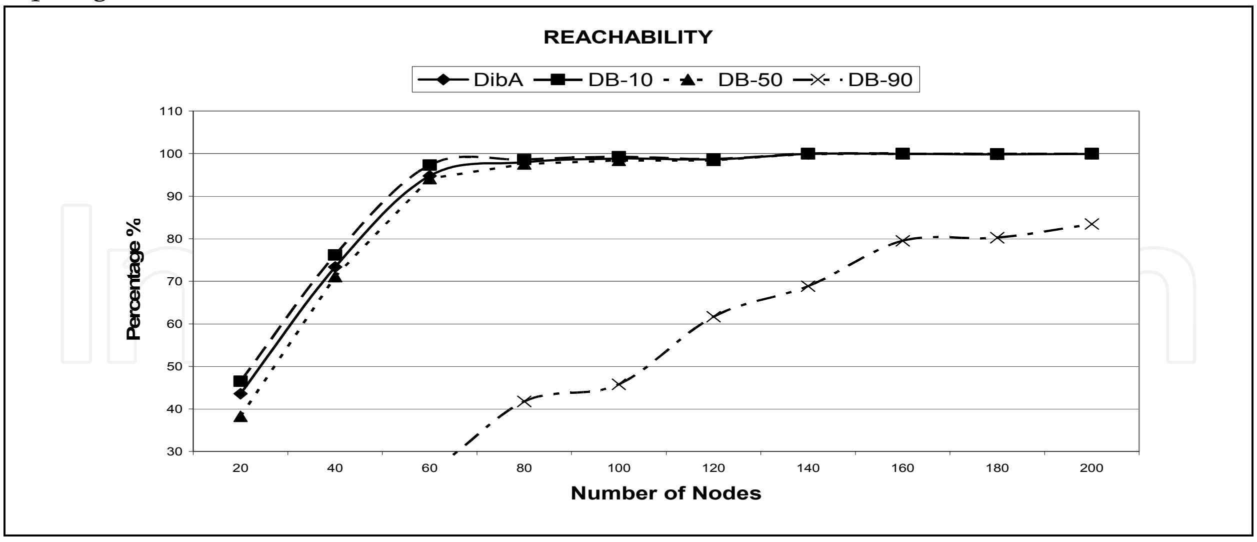


Fig. 11. Reachability – Normal – BGR 5p/s

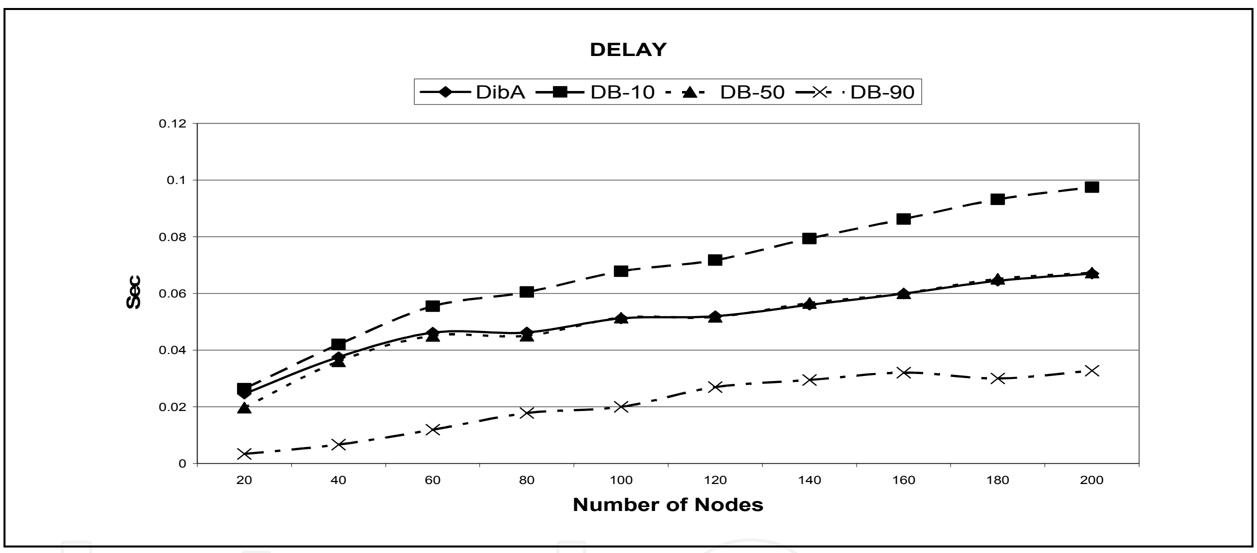


Fig. 12. Delay – Normal – BGR 5p/s

in order to cover the majority of possible scenarios. The network area is of fixed size $500x500m^2$. The mobility files are created with zero pause time. Mobile nodes move with maximum speed of 5m/sec. Each simulation has duration of 100secs and all mobile nodes use a transmission range of 100m.

Each scenario is restricted to the transmission of broadcast traffic only. This is a common strategy, especially when using very high broadcast generation rates (BGR). Combining normal traffic with broadcast traffic is a step further for our work with the implementation currently taking place. In order to avoid anomalies, we run three simulations for every scenario using three different mobility files. Our research has found no work until this point, where more than 3 or 4 repetitions are used. The final results are created as an average of the three simulations.

Experiments where performed using 3 different distance thresholds for DB of 10m, 50m and 90m, in order to cover the two extremes and an intermediate value. DibA tunes the distance threshold to one of the 3 thresholds mentioned above, depending on the local level of density. The number of nodes has a starting value of 20 and reaches a maximum of 200 nodes with a step of 20 (20, 40, 60, …, 200).

We first divide our simulations into two groups according to the broadcast generation rate. BGR is set to 5packets/sec and 60packets/sec. Furthermore, we also divide the simulations depending on whether a normal or a diverse topology is used.

The following performance metrics are considered:

- Reachability – The percentage of nodes that successfully receive the broadcast message.
- Delay – The time elapsed from the initiation of the broadcast process until no more rebroadcasts take place.
- Average number of Packets transmitted per node (APT) – This is a self explained performance metric which is closely related to energy efficiency.

5.2 Simulation results

Fig. 11, 12 and 13 present the performance of the 4 schemes, when normal scenarios are used and BGR is set to 5packets/sec.

Fig. 11 shows that DB-90 performs very poorly due to the high threshold value, whereas all the other schemes perform almost identical. Although DB-90 appears to be very fast in Fig. 12, that is because of the very low level of reachability. DB-10 is the slowest, despite the fact that has similar reachability with DB-50 and DibA. The latter two again perform in a similar way. Fig. 13 shows that DB-10 uses a significantly higher number of transmissions in order to achieve the same level of reachability with DB-50 and DibA. Thus, it is the least energy efficient.

Fig. 14, 15 and 16 show how the 4 schemes perform when the topology is diverse and the broadcast generation rate is low.

Fig. 14 reflects the performance of all schemes in terms of reachability. Although DibA, DB-10 and DB-50 perform almost identical when the network is dense (120 nodes or more), for sparse topologies DB-10 is slightly better than DibA and in turn that is better than DB-50. DB-90 again performs poorly. DB-10's slightly better performance for reachability, proves to be extremely costly, as it is much slower than the rest and APT is almost double than the

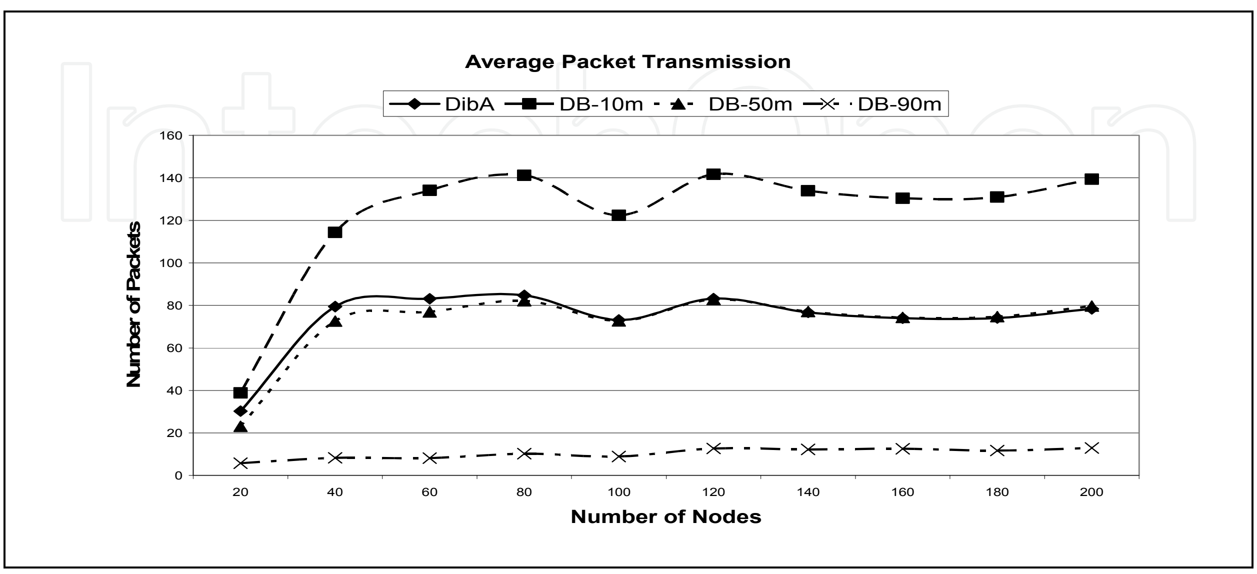


Fig. 13. APT – Normal – BGR 5p/s

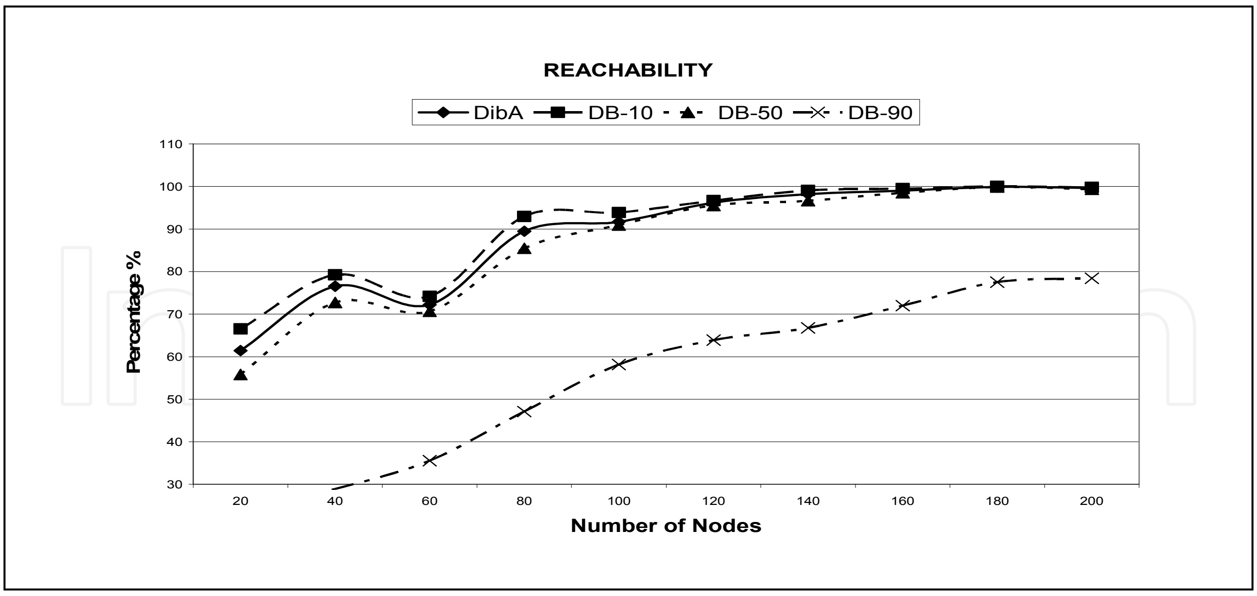


Fig. 14. Reachability – Diverse – BGR 5p/s

following scheme. Energy efficiency is very poor in these conditions. DibA appears to be better than DB-50 for sparse topologies and similar when density increases. Better reachability usually comes with more latency and more APT. For DibA and DB-50 this is reflected in Fig. 15 and 16.

Fig. 17, 18 and 19 present the performance of the 4 schemes when normal scenarios are used and BGR is set to 60packets/sec.

Fig. 17 shows that for sparse networks (up to 60 nodes) DibA and DB-10 have the same performance with DB-50 being slightly worse. For very dense networks, DibA is now performing better than the rest. DB-90 is completely outperformed. Despite the fact that DB-10 has lower reachability when compared to DibA, Fig. 18 and 19 show that it is disproportionally slower and energy inefficient. DB-50 shows slightly better performance for delay and APT, but that is due to its lower reachability.

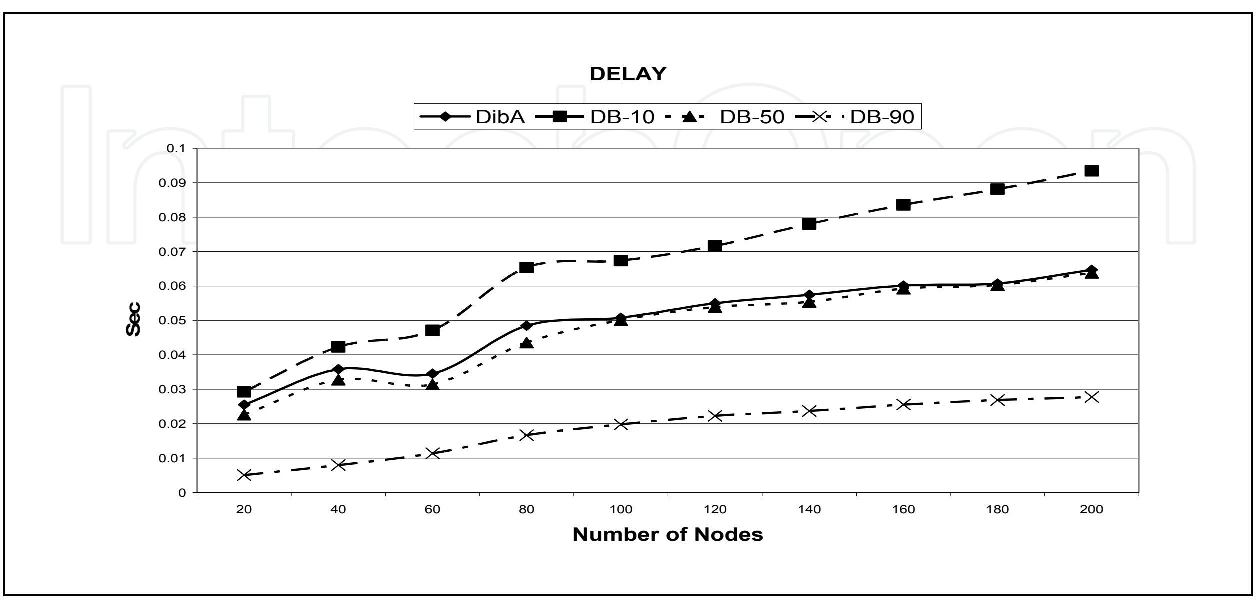


Fig. 15. Delay – Diverse – BGR 5p/s

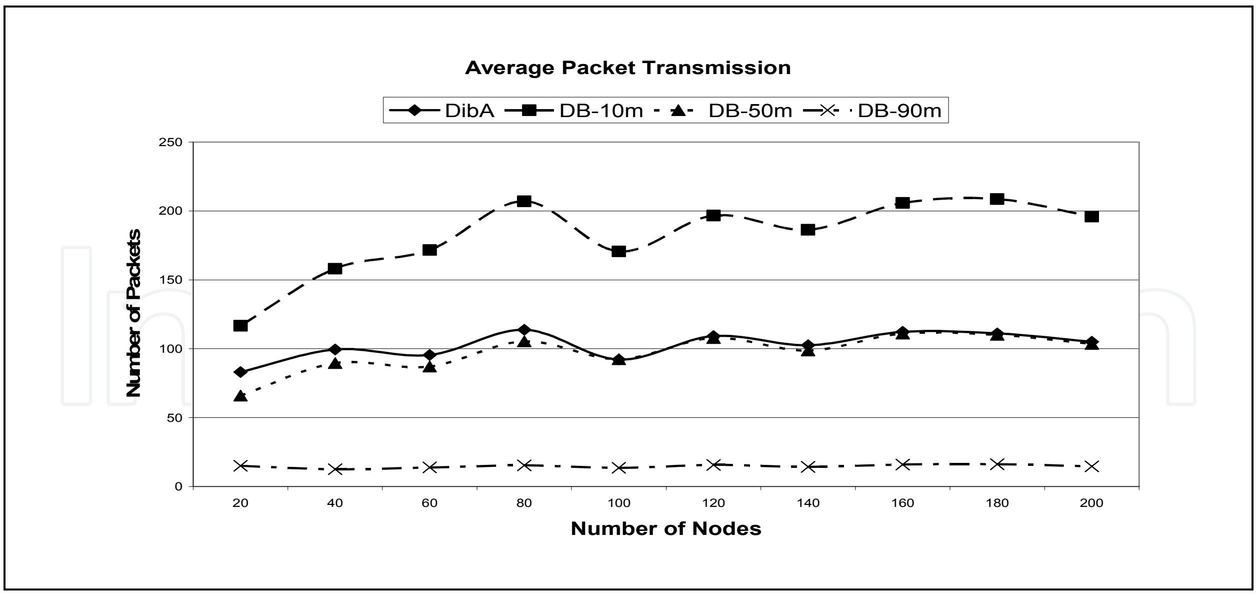


Fig. 16. APT – Diverse – BGR 5p/s

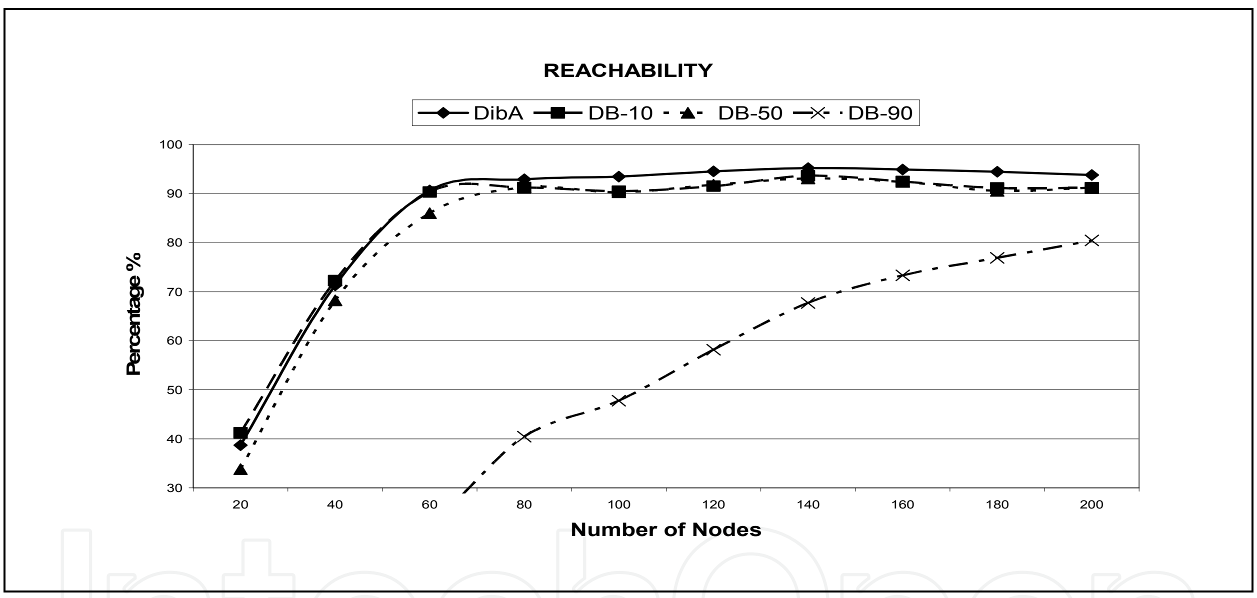


Fig. 17. Reachability – Normal – BGR 60p/s

Fig. 20, 21 and 22 show how the 4 schemes perform when the topology is diverse and the broadcast generation rate is extremely high.

In this group of experiments we have used a very high broadcast generation rate and extremely diverse network topologies. The results, in terms of reachability, are reflected in Fig. 20. DB-10 is better for sparse networks, but as density increases, it is found to finish last for dense networks of 200 nodes. DB-50 proves to be more stable, but at no point does it perform better than all the rest. The results for DB-90, prove that even the use of a very low distance threshold is the appropriate selection when both density and traffic rate are set to high values. DibA appears to be the most reliable option. Fig. 21 and 22 show that DB-10 is neither fast nor energy efficient. DB-50 performs well but being faster and more energy efficient is the result of its low reachability levels.

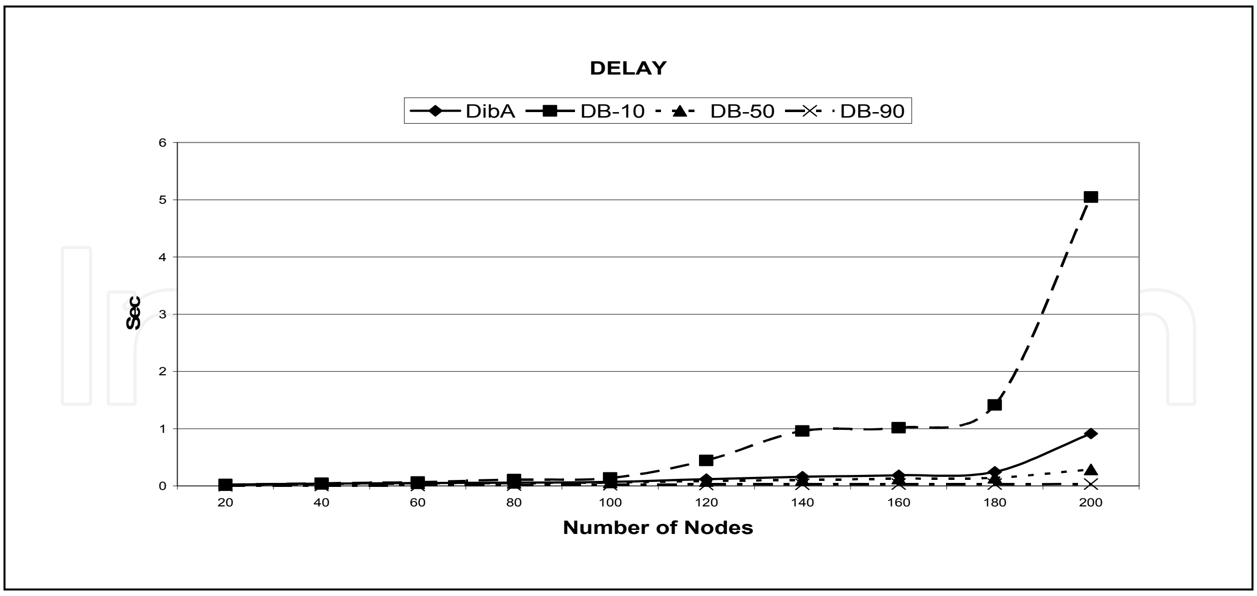


Fig. 18. Delay – Normal – BGR 60p/s

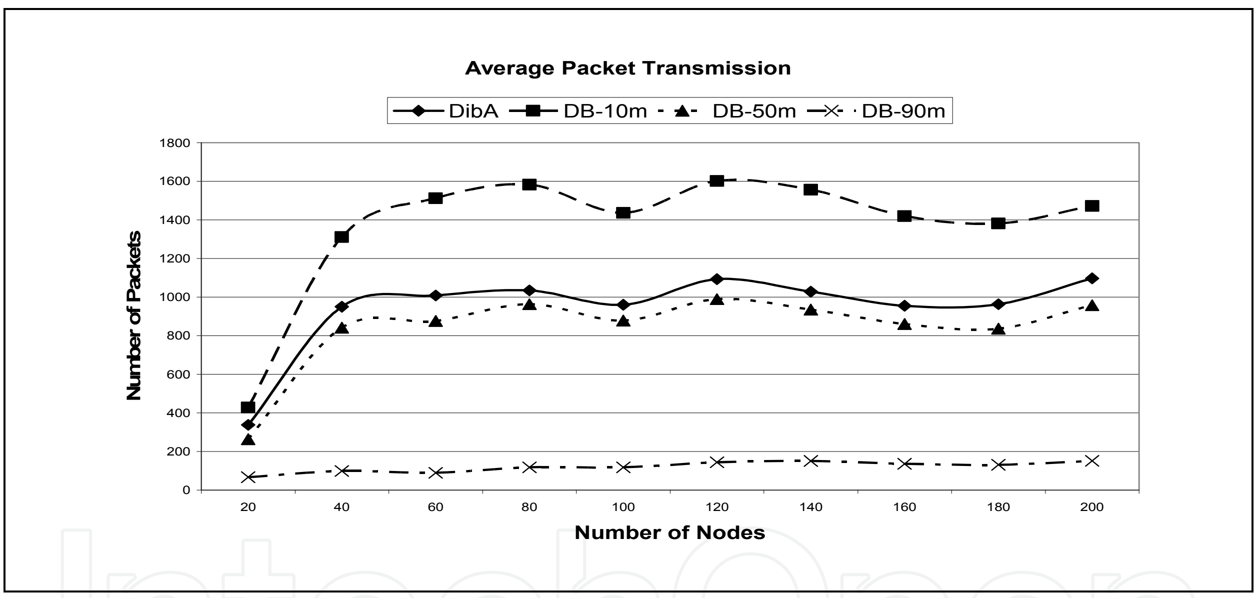


Fig. 19. APT – Normal – BGR 60p/s

6. Conclusions

In this chapter we have shown how the Distance-Based broadcast scheme can potentially alleviate the effects of Simple Flooding by controlling the amount of replicated messages. We have also demonstrated how the selection of a single, fixed and pre-determined distance threshold is not appropriate for all scenarios and does not satisfy the needs of highly dynamic network topologies despite the fact that the Broadcast Storm Problem was overcome when SF was substituted by DB. When the network is sparsely populated and the broadcast generation rate is also low, a small distance threshold needs to be selected in order for the broadcast operation not to "die out" due to lack of pathways to remote or isolated mobile nodes inside the topology. Respectively, a large distance threshold is

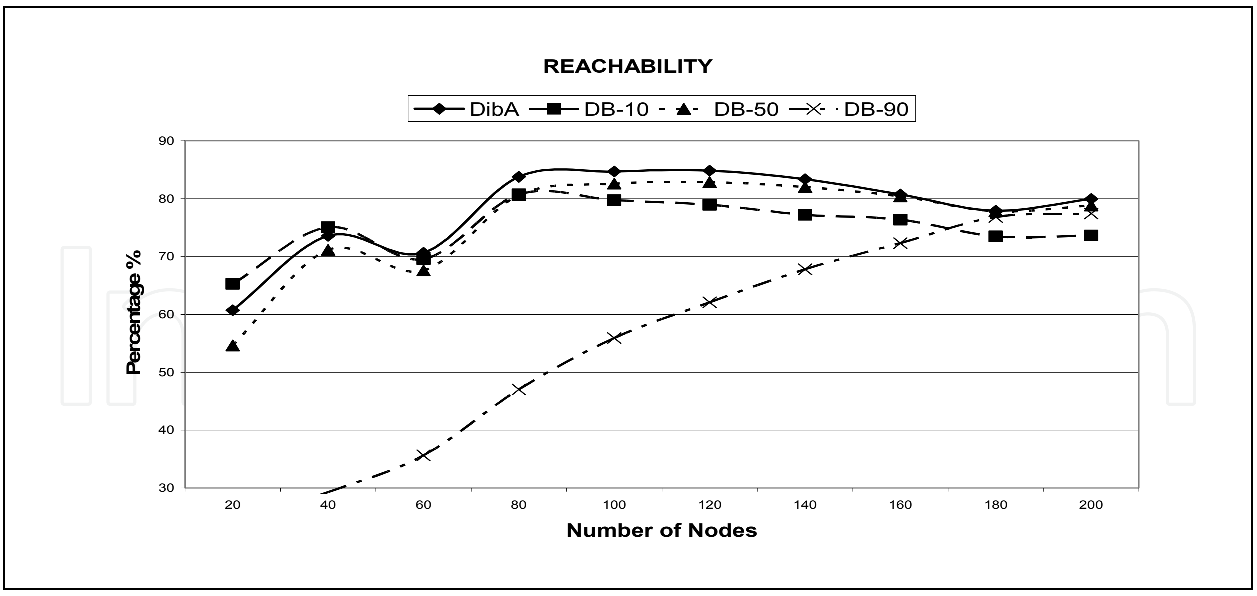


Fig. 20. Reachability – Diverse – BGR 60p/s

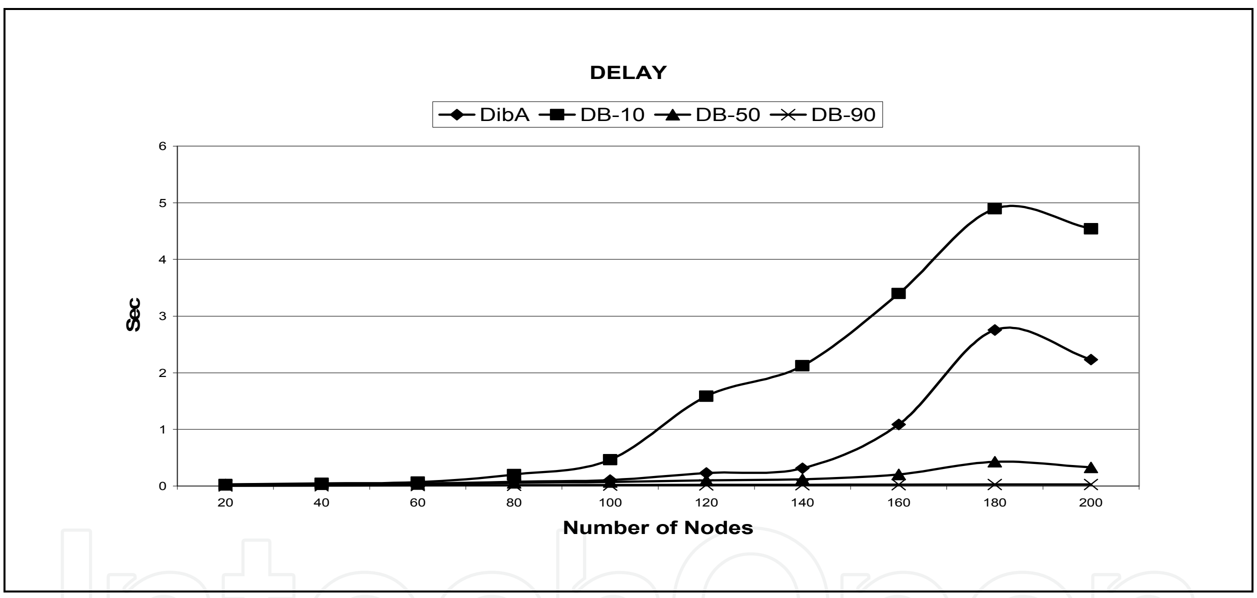


Fig. 21. Delay – Diverse – BGR 60p/s

required for very dense topologies with high broadcast generation rates as it requires fewer broadcast relays to take place decreasing in that way the volumes of contention and collisions. Consequently, adaptive schemes were proposed that adjust the thresholds used depending on the local density of the network.

The adaptive schemes proposed so far introduce further overhead. Alternatively, schemes that make use of positioning systems require the existence of such expensive and unreliable hardware, e.g. GPS. In this chapter, we have presented a new scheme, called DibA, utilising duplicate packets to adjust the distance threshold accordingly. A small threshold is individually set for every node when the duplicate packet counter is small in order to force more nodes to rebroadcast. Respectively, a large distance threshold is set when the counter is increased to a high value aiming in fewer broadcast relays.

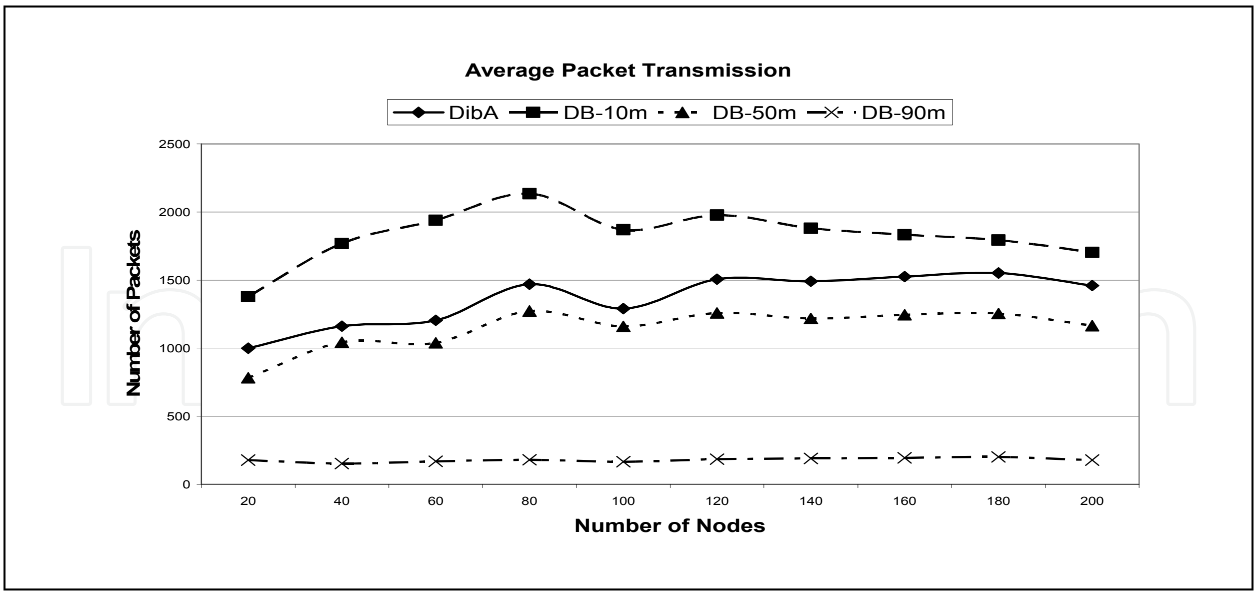


Fig. 22. APT – Diverse – BGR 60p/s

Our performance study compared DibA against DB with three different distance thresholds of 10m, 50m and 90m under common and diverse operational conditions throughout simulation. In order for our study to cover all extremes, we have also made use of highly diverse network topologies that include both sparse and dense areas. These scenarios have aided in demonstrating the superiority of DibA over DB more appropriately. In all scenarios, the simulation results clearly showed that DibA outperforms DB for various topologies and broadcast generation rates. Furthermore, DibA is also more reliable and power efficient than DB as the number of broadcast relays does not linearly grow up with the network density.

7. References

D. Cavin, Y. Sasson & A. Schiper. (2002). On the Accuracy of MANET Simulators, *Proceedings of POMC*, pp. 38-43, Toulouse France, 2002, ACM, New York, USA.

K. Viswanath & K Obraczka. (2006). Modeling the performance of flooding in wireless multi-hop ad hoc networks, *Computer Communications*, Vol. 29, No. 8, pp. 949-956.

S. Ni, Y. Tseng, Y. Chen & J. Sheu. (1999). The broadcast storm problem in a mobile ad hoc network, *Proceedings of MOBICOM*, pp. 151-162, Seatttle, Washington, USA, 1999, ACM, New York, USA.

S. Leng, L. Zhang, L. Wu Yu & C. Heng Tan. (2004). An efficient broadcast relay scheme for MANETs, *Computer Communications*, Vol. 28, No. 5, pp. 467-476, Butterworth-Heinemann, Newton, MA, USA.

C. Zhu, M. J. Lee & T. Saadawi. (2004). A Border-aware broadcast scheme for Wireless Ad Hoc Networks, *Proceedings of CCNC*, pp. 134-139, 2004, IEEE.

A. Qayyum, L. Viennot & A. Laouiti. (2002). Multipoint relaying for flooding broadcast messages in mobile wireless networks, *Proceedings of HICSS*, pp. 298, 2002, IEEE Computer Society, Washington, DC, USA.

C. Hsu, C. Chen & H. Wang. (2005). DISCOUNT: A Hybrid Probability-Based Broadcast Scheme for Wireless Ad Hoc Networks, *Proceedings of VTC-FALL*, pp. 2706-2710, Dallas, USA.

R. Purtoosi, H. Taheri, A. Mohammadi & F. Foroozan. (2006). Improving broadcast performance by traffic isolation in wireless ad hoc networks. *International Journal of Communication Systems*, Vol. 19, No. 9, pp. 1029–1043.

B. J. Barritt, B. Malakooti & Z. Guo. (2006). Intelligent Multiple-Criteria Broadcasting in Mobile Ad-hoc Networks, *Proceedings of IEEE LCN / P2MNet*, pp. 761-768, Tampa, Florida, USA.

N. Bauer, M. Colagrosso & T. Camp. (2005). Efficient implementations of all-to-all broadcasting in mobile ad hoc networks, *Pervasive and Mobile Computing*, Vol. 1, No. 3, pp. 311–342.

B. Williams & T. Camp. (2002). Comparison of broadcast techniques for mobile ad hoc networks, *Proceedings of MOBIHOC*, pp. 194-205, Lausanne, Switzerland.

Young-Ching Deng, Ching-Chi Shue & Ferng-Ching lin. (2006). An Adaptive Medium Access Control Protocol for Reliable Broadcast and Unicast in Ad Hoc Networks, *IEICE – Transactions on Information & Systems*, pp. 527-535.

Jung-Pil Ryu, Min-Su Kim, Sung-Ho Hwang & Ki-Jun Han. (2004). An Adaptive Probabilistic Broadcast Scheme for Ad-Hoc Networks, In: *High Speed Networks and Multimedia Communications*, pp. 646-654, Springer Berlin / Heidelberg, Toulouse, France, 2004.

Sung-Hee Lee & Young-Bae Ko. (2006). An Efficient Neighbor Knowledge Based Broadcasting for Mobile Ad Hoc Networks, In: *Computational Science - ICCS*, pp. 1097-1100, Reading, UK.

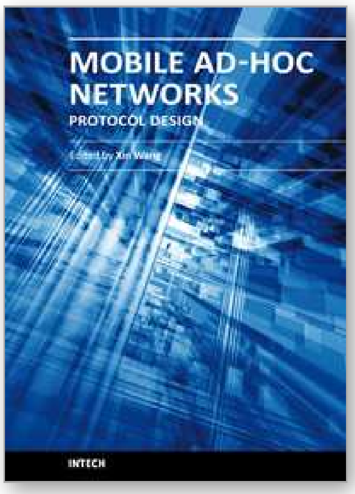
Xiaohu Chen, Michalis Faloutsos & Srikanth Krishnamurthy. (2002). Distance ADaptive (DAD) Broadcasting for Ad Hoc Networks, *Proceedings of MILCOM*, pp. 878-882.

Michael D. Colagrosso. (2007). Intelligent Broadcasting in Mobile Ad Hoc Networks: Three Classes of Adaptive Protocols, *Proceedings of EURASIP*, pp. 25-25.

Xiaohu Chen, Michalis Faloutsos & Srikanth V. Krishnamurthy. (2003). Power Adaptive Broadcasting with Local Information in Ad hoc networks, *Proceedings of ICNP*, pp. 168-178.

Pradeep Kyasanur, Romit Roy Choudhury & Indranil Gupta. (2006). Smart Gossip: An Adaptive Gossip-based Broadcasting Service for Sensor Networks, *Proceedings of MAAS*, pp. 91-100, Vancouver, BC, Canada.

Yu-Chee Tseng, Sze-Yao Ni & En-Yu Shih. (2001). Adaptive Approaches to Relieving Broadcast Storms in a Wireless Multihop Mobile Ad Hoc Network, *Proceedings of Distributed Computing Systems*, pp. 481-488.



Mobile Ad-Hoc Networks: Protocol Design
Edited by Prof. Xin Wang

ISBN 978-953-307-402-3
Hard cover, 656 pages
Publisher InTech
Published online 30, January, 2011
Published in print edition January, 2011

Being infrastructure-less and without central administration control, wireless ad-hoc networking is playing a more and more important role in extending the coverage of traditional wireless infrastructure (cellular networks, wireless LAN, etc). This book includes state-of-the-art techniques and solutions for wireless ad-hoc networks. It focuses on the following topics in ad-hoc networks: quality-of-service and video communication, routing protocol and cross-layer design. A few interesting problems about security and delay-tolerant networks are also discussed. This book is targeted to provide network engineers and researchers with design guidelines for large scale wireless ad hoc networks.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821