Reliable and Efficient Broadcasting in Vehicular Ad Hoc Networks

Francisco J. Ros
DIIC, University of Murcia
Murcia, Spain E-30100
Email: fjrm@dif.um.es

Pedro M. Ruiz
DIIC, University of Murcia
Murcia, Spain E-30100
Email: pedrom@dif.um.es

Ivan Stojmenovic
SITE, University of Ottawa, Canada
EECE, University of Birmingham, UK
www.site.uottawa.ca/∼ivan
Email: stojmenovic@storm.ca

***Abstract*—Most of the envisioned services over vehicular networks need to deliver information to all vehicles inside a certain region. Several such broadcasting protocols have been reported so far, but surprisingly only one of them addresses the issue of intermittent connectivity.**

In this paper, we present a broadcast protocol which is suitable for a wide range of vehicular scenarios and traffic conditions. The protocol employs local position information acquired via periodic beacon messages. Beacons are used by cars to decide whether or not they belong to a connected dominating set (CDS). Vehicles in the CDS use shorter waiting period before possible retransmissions. Identifiers of circulated broadcast messages are added to beacons as piggybacked acknowledgements. When waiting timeout expires, vehicle retransmits if it has at least one neighbor which did not acknowledge circulated message with the last beacon, and sets a new waiting period. Our algorithm does not depend on any parameter or threshold which varies its operation. Despite its simplicity, the protocol is shown to provide high reliability and efficiency by means of a simulation-based performance evaluation. It also greatly outperforms the only competing algorithm we found in the literature which explicitly considers different mobility scenarios.

I. INTRODUCTION AND MOTIVATION

Vehicular communications are currently a hot research topic because of the plethora of services which are expected to be developed over them. Envisioned applications range from critical safety services and advanced driver assistance systems (ADAS) to traffic management and infotainment applications [1]. Most of these services rely on the delivery of broadcast messages to the vehicles inside certain area of interest. Therefore, the design of a reliable and efficient broadcast protocol is of paramount importance for the successful deployment of vehicular communication services.

Several protocols have been previously proposed for broadcasting in VANET. However, they are designed for either rectilinear highways/roads ([2][3][4]) or urban grids ([5][6]). More surprisingly, they do not address the issue of temporary disconnections in VANET, which is one of its most salient properties. The Distributed Vehicular Broadcast (DV-CAST) protocol [7] is the only solution we found in the literature which explicitly addresses the various connectivity conditions which are present in vehicular networks. Unfortunately, it can only be applied to simple rectilinear streets with several lanes. Vehicle behavior is decided by its status. It is in *well-connected* status if it has at least one neighbor of the same cluster in the message forwarding direction. In such case, the well-connected vehicle runs one of the broadcast suppression techniques described in [8]. A vehicle is operating in sparsely-connected regime if it is the last one in a cluster of vehicles. In addition, it is said to be in a *sparsely-connected neighborhood* if it has at least a neighbor in the opposite direction. Otherwise, the vehicle is in a *totally disconnected neighborhood*. Upon receiving a packet, the sparsely-connected vehicle immediately rebroadcasts it. If it moves in the same direction as the original message source, the packet is then discarded. Otherwise the packet is carried until it expires or can be retransmitted back to the original message forwarding direction. Message is carried afterwards until an implicit acknowledgement is received (from another vehicle with greater hop count), and is being retransmitted in the meanwhile if new neighbors are identified. A vehicle in totally disconnected mode carries the message until a new neighbor is identified, retransmits it with probability 1 immediately, and discards it afterwards.

In this paper, we propose a fully-distributed adaptive algorithm suitable for different vehicular scenarios and traffic conditions. It does not depend on threshold values nor different internal states which vary the protocol behavior. Each vehicle decides by itself whether to forward a received broadcast message or not. Such decision is solely based on the local information which the vehicle has acquired from its neighborhood by means of periodic beacon messages. This guarantees ultimate scalability. In our implementation, beacons contain the position of the sender and (additionally) identifiers of the recently received broadcast messages, which serve as acknowledgements of reception. Applying a heuristic for computing *connected dominating sets* (CDS) and the *neighbor elimination scheme* (NES) [9] (augmented with the received acknowledgements), the protocol makes the forwarding decision trying to minimize the number of transmissions while maximizing the reliability of the broadcast task. Our proposal is an extension and adaptation of the *parameterless broadcast in static to highly mobile* (PBSM) ad hoc networks protocol [10]. Our target is to improve the protocol efficiency in VANET scenarios. The major changes and improvements made include using broadcast message acknowledgements to avoid redundant retransmissions and improve reliability, and employing 1-hop position information instead of 2-hop neighbor information which is used in [10].

978-1-4244-2517-4/09/$20.00 ©2009 IEEE

The algorithm [10] requires adding the list of all neighbors to the beacons, increasing message size and leading to the increase of collisions and message failures. The algorithm described here makes use of acknowledgements, including identifiers of recently broadcasted packets in beacon messages.

By means of a simulation study with different vehicular scenarios under different traffic conditions, we show that the solution achieves high reliability with few redundant transmissions. Compared to the original PBSM, the proposed scheme maintains the same reliability while highly improving the protocol efficiency in terms of the number of needed retransmissions. Furthermore, we implement and simulate the DV-CAST protocol [7]. Contrary to our proposed algorithm, DV-CAST changes the protocol behavior depending on the traffic regime sensed by the corresponding vehicle. Our simulation results demonstrate that our proposal greatly outperforms DV-CAST in all evaluated scenarios.

The remainder of this paper is organized as follows. Section II summarizes the techniques and protocols that our solution is built upon. Section III describes the proposed algorithm in detail. The simulation-based performance evaluation and related discussions are presented in Section IV. Finally, Section V concludes the paper.

II. RELATED WORK

Due to space constraints, we briefly review the techniques and protocols which are directly related to our proposed solution. A common technique to reduce redundant transmissions in a broadcast task is the use of *connected dominating sets* (CDS). A subset $V_D \subseteq V$ is said to be dominating if each node in V either belongs to V_D or has at least one neighbor which belongs to V_D. V_D is a CDS if it is connected. In CDS-based broadcasting, only nodes belonging to the CDS are retransmitting the broadcast message, which suffices for packet to reach the whole network. Therefore, the fewer number of nodes in the CDS, the less redundancy in the broadcast protocol. Unfortunately, the problem of finding the minimum CDS was shown to be NP-hard [11]. Among the various heuristics which have been proposed to compute a CDS, we focus on the one described in [9]. There, each node decides, based on the local information obtained via beacon messages, its own status according to the following definitions. A node is said to be *intermediate* if it has at least two 1-hop neighbors which are not directly connected. A node u is covered by a neighbor node v if any other neighbor of u is also a neighbor of v, and if v has higher key than u. Intermediate nodes which are not covered by any neighbor are called *intergateway* nodes. Finally, intergateway nodes which are not covered by any pair of connected 1-hop neighbors become *gateway* nodes. A CDS made of gateway nodes generates small sets and therefore is an efficient broadcast structure [9]. We have used vehicles' unique identifiers as keys.

The former technique is further improved when the *neighbor elimination scheme* (NES) is employed. When a broadcast message is received, it is not immediately forwarded. Instead, the node sets up a waiting timeout and monitors its neighborhood. When the timeout expires, the message is retransmitted only if the node still has uncovered neighbors [9].

The *parameterless broadcast in static to highly mobile* (PBSM) ad hoc networks protocol [10] makes use of the aforementioned techniques to develop an adaptive algorithm which does not depend on any parameter or threshold value. Because of its flexibility and good performance, it is used as the basis of the protocol proposed in this paper for vehicular ad hoc networks. In PBSM, each vehicle S maintains two lists of neighboring cars with respect to the message being disseminated and local 1-hop knowledge: R and N, containing neighbors that already received (did not receive, respectively) the message. After a delay timeout, S retransmits the message if the list N is nonempty. Both lists R and N are updated with every copy of message and beacon exchange received, which may trigger further retransmissions if N becomes again nonempty. Nodes in CDS set shorter waiting timeouts than nodes that are not in CDS.

III. PROPOSED PROTOCOL

A. Overview

The main problem that a broadcast protocol for VANET must face is its adaptability to the different vehicular scenarios. It should achieve high coverage of the network at the expense of as few transmissions as possible, regardless whether the network is extremely dense (e.g. big cities at rush hours) or highly disconnected (e.g. highways at night). It should be also functional in the presence of cars with variety of speeds simultaneously: static at traffic lights, moderate or high mobility, or different speeds in different lanes.

Our proposal is an extension of PBSM which tries to reduce the protocol redundancy under the aforementioned vehicular situations. The main novelty is the modification of the algorithm to handle acknowledgements of broadcast messages. Such acknowledgements are piggybacked in periodic beacons. Hence, the protocol is called *Acknowledged PBSM* (AckPBSM). Although PBSM was defined with 2-hop topological information, it can also work with 1-hop position information. We chose this last alternative for AckPBSM because it introduces lower overhead in beacon messages. Like PBSM, our solution employs the store-carry-forward paradigm to deliver the message in highly partitioned networks. So, every vehicle stores the broadcast message in a buffer until its expiration. AckPBSM protocol constructs a CDS delivery backbone with the information obtained in the beacon exchange phase, as in PBSM. However, since broadcast messages are acknowledged, new discovered neighbors which previously received the message do not cause new retransmissions.

Let us see an example of the operation of AckPBSM before giving the protocol details. Given the scenario depicted in Figure 1, vehicle A generates a broadcast message which is first buffered by A (in case new neighbors without the message are found in the future), and then received by B, C, D. Receivers set up a waiting timeout which is shorter if the vehicle belongs

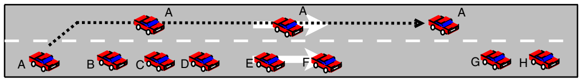

Fig. 1. Common vehicular scenario.

to the computed CDS. Let D be in the CDS, so it retransmits first. B and C cancel their retransmission because all their neighbors have been covered by D's forwarding.Vehicles E and D receive the message. However, none of them have uncovered neighbors, so the transmission does not take place. Along this process, receivers acknowledge the reception of the message. In case any vehicle fails to receive the message, its neighbors detect the situation because of the lack of acknowledgement and repeat the former steps in order to cover it.

Now, let us assume that A speeds up and overtakes vehicles $B-F$. Although A and E, F were not neighbors previously, no new transmission occurs because the acknowledgements in beacons clarify that none of those vehicles need the message. This is not the case for other protocols which do not consider acknowledgements for broadcast messages, like PBSM, and therefore they would cause unnecessary retransmissions. Finally, A contacts G, which is not acknowledging the reception of the message. Then, A can transmit the message that it carried for a while. If G has an uncovered neighbor H, a new transmission occurs and the whole network gets covered. In the following, the algorithmic details to achieve the described protocol are set forth.

B. Protocol details

Vehicles issue periodic beacon messages in order to gain knowledge about the local network topology. After each beacon exchange, this information is used to determine whether or not the vehicle itself is part of the CDS. In addition, beacons are augmented with the identifiers of the broadcast messages which have been received recently. These acknowledgements are included during the next H seconds since the reception of the first copy of the message.

Let X be the vehicle under consideration. For each broadcast message, vehicles set up two lists: R and N. R consists of those neighbors of X which are supposed to have received the message (based on local topology knowledge of X). N contains the remaining 1-hop neighbors of X.

There exists a timeout function to_{ev} which assigns waiting time to each vehicle before possible retransmission. The value returned by to_{ev} can be proportional to $1/|N|$, being $|N|$ the number of elements in N, and depends on whether the node is currently in the CDS or not (shorter waiting time if in CDS).

In addition, whenever a new neighbor (except the source of a newly received message) is inserted into R, X initializes a timeout to_{ack} attached to such neighbor. It is used to wait for the acknowledgement of reception. The timeout value is a constant larger than beacon interval (since acknowledgements are received via beacons). If to_{ack} expires and acknowledgement is not received with the next beacon, the corresponding neighbor is moved from R to N, or removed if expected beacon is not received. This may cause the reactivation of to_{ev} if N was empty before the insertion of the new element. If to_{ev} was already running, it could also be updated in case it depends on $|N|$.

Data source adds all its known neighbors to R and transmits the generated broadcast message. Corresponding to_{ack} timers are initialized. Upon receiving the broadcast message, X includes in R the sender and all its known neighbors (and starts to_{ack} timers), because they should have also received the message. Accordingly, those vehicles are removed from N. The remaining neighbors of X which are not connected to the sender are inserted into N. In case N becomes empty ($|N| = 0$), X cancels to_{ev} and decides not to retransmit. In effect, this means that all the neighbors of X have been already covered by the retransmissions of other vehicles. Otherwise, if $|N| > 0$ and to_{ev} depends on $|N|$, it is updated. When to_{ev} expires, if N is not empty then X retransmits the message and moves the content of N to R (causing the activation of timeouts to_{ack}).

X removes from N nodes with missing expected beacon, thus declaring them as disconnected. If newly reported 1-hop neighbors are not in R, they are added to N (causing reactivation of to_{ev} if N was empty). For each acknowledged message listed within the beacon from neighbor B, X cancels the associated to_{ack} and adds/confirms B in R (removing it from N if it was there).

C. Discussion

Our approach is appropriate for vehicular scenarios such as urban layouts with intersections. Those vehicles located at junctions which are the only ones with connectivity with other vehicles at converging streets, will be selected as dominating and therefore will retransmit to propagate the broadcast message along those streets.

By piggybacking the acknowledgements in beacon messages, each vehicle can update its lists R and N accordingly. The objective is twofold: reduce redundancy and improve reliability in the advent of message losses.

Obviously, the longer the time H during which the given message is being periodically acknowledged, the fewer redundant transmissions will occur. By setting H equal to the message lifetime, the protocol behaves well by acknowledging the message only during the time it is carried. This feature is important in vehicular scenarios due to their particular mobility patterns. As we saw in Figure 1, vehicle A discovers new neighbors E, F as it is overtaking. New transmissions occur in the case of PBSM, since new neighbors must be covered. However, these are redundant because all the vehicles already received the message. AckPBSM saves redundant transmissions because the beacons contain the acknowledgement of the message, and therefore the new discovered neighbors are not always covered. This situation is common when vehicular mobility patterns are considered, and also occurs in other similar settings (e.g. when a slow vehicle is being overtaken by a cluster of fast vehicles, when vehicles leave a cluster and join it later on due to stops at intersections or traffic lights, etc.).

There are many factors which influence the reception of a wireless signal (attenuation, multi-path fading, interference, etc.). Thus, a vehicle B, which has been sensed as a neighbor because of the reception of its beacons, might not correctly decode the forwarded message. In such case, B's neighbors will not receive the corresponding acknowledgement and AckPBSM would consider it again for coverage by a retransmission.

Contrary to protocols like DV-CAST, our solution does not need to determine the traffic regime that is sensed by the vehicle. This simplicity is a great advantage: since there are no different internal states, flaws due to unexpected situations are less prone to appear.

IV. Performance Evaluation

We have implemented the AckPBSM, PBSM and DV-CAST protocols in *The Network Simulator ns-2*[1], version 2.33. The *SUMO* tool[2] has been employed to create two scenarios, highway and urban, and to generate the mobility traces of the simulated vehicles. This allows us to simulate common vehicular situations such as overtakes and stops at intersections. This leads to intermittent connectivity and uneven distribution of vehicles. In each scenario, we defined several routes which are followed by the vehicles. SUMO injects cars in each route according to a given traffic rate, measured in vehicles injected per second. In order to get a wide range of network connectivity, we have varied the traffic injection rate per route from $1/75$ to $1/5$ (highway) and $1/15$ (urban) vehicles per second. The higher the traffic injection rate, the higher the network density. In the urban scenario we defined more routes, so that a lower rate ($1/15$) generates a network density comparable to the highway set-up with higher rate ($1/5$). The figures and tables in this section are labeled with the reciprocal of this rate, i.e., with the interval between the injection of consecutive vehicles (from 75 to 5 seconds). Two kind of cars have been defined, with maximum speeds of 50 and 80 km/h.

Although PBSM was not specifically designed for vehicular networks, we include it in the performance evaluation to measure the improvement that AckPBSM achieves over it. Both AckPBSM and PBSM implement the CDS heuristic described in [9] with 1-hop position information. In AckPBSM, the reception of a message is acknowledged while such message remains in the vehicle's buffer ($H = 30$ sec). Timeout to_{ev} is computed as $0.25/|N|$ seconds if the node is in CDS, $0.25 \cdot (1 + 1/|N|)$ otherwise. to_{ack} is fixed to 1.6 seconds. Regarding DV-CAST, we implemented the weighted p-persistence algorithm as the broadcast suppression technique. The other slot-based approaches were not chosen because they, as recognized by the authors, depend on parameters which may be hard to tune in practice [8].

TABLE I

Simulation parameters.

Traffic Rate	(1/75, 1/60, 1/45, 1/30, 1/15, 1/5) veh/sec/route
Message Lifetime	30 sec in buffer
Simulation Time	120 sec (after steady state)
No. of Runs	20 (95% confidence interval shown)
Beacon Interval	0.5 sec
Beacon Hold Time	1.5 sec

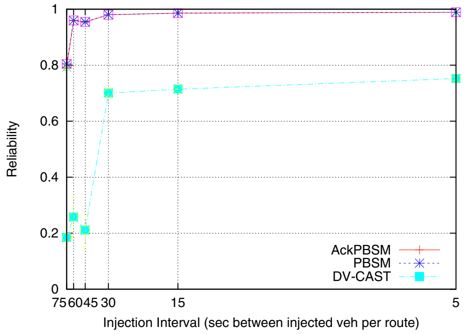

Fig. 2. Reliability in the highway scenario. 95% confidence interval drawn along the y-axis.

Simulated vehicles use an ideal MAC layer in which no collisions occur. Wireless signals follow the two-ray-ground propagation model, with a fixed transmission range of 250 meters. These ideal conditions allow us to concentate on the broadcast protocols themselves, although we also conducted simulations with the IEEE 802.11p MAC layer and the results were analogous (not shown due to space constraints).

Each run consists of one broadcasting task (500 bytes message). Table I summarizes the simulation setup.

We focus on two different metrics:

- Reliability. Defined as the ratio between the number of vehicles which receive the broadcast message and the total number of simulated vehicles: $Rel = N_{recv}/N_{total}, Rel \in [0, 1)$. The reliability may not reach 100% because some cars may remain partitioned from the source.
- Number of transmissions per transmitting vehicle. This measures the efficiency of the protocol.

A. *Highway scenario*

This scenario consists of a 4000 meters long rectilinear highway with two lanes per direction. As shown in Figure 2, AckPBSM and PBSM offer the same reliability regardless the flow rate under consideration. For sparse networks (traffic rate from $1/75$ to $1/60$), the broadcast message is delivered from the 80% to the 95% of the total number of vehicles. It virtually covers the whole network for regular and dense traffic conditions. On the other hand, DV-CAST offers a very poor reliability for sparse networks, while it only covers the 75% of vehicles with the highest traffic rate. The reason is that the protocol does not foresee common vehicular movements such as passing maneuvers. For example, let us look again to Figure 1. Assume that vehicle F initiated the broadcasting

---

[1] http://www.isi.edu/nsnam/ns/

[2] http://sumo.sourceforge.net/

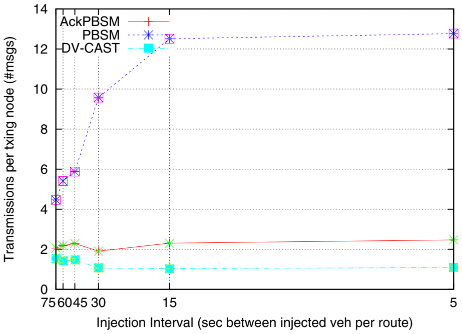

Fig. 3. Transmissions per vehicle in the highway scenario. 95% confidence interval drawn along the y-axis.

TABLE II
RELIABILITY RESULTS IN THE URBAN SCENARIO FOR DIFFERENT INJECTION INTERVALS (SEC BETWEEN INJECTED VEH PER ROUTE).

Interval	75	60	45	30	15
AckPBSM	0.313	0.559	0.933	0.991	0.999
PBSM	0.295	0.673	0.939	0.994	0.999
DV-CAST	0.188	0.278	0.506	0.553	0.922

TABLE III
TRANSMISSIONS PER VEHICLE IN THE URBAN SCENARIO FOR DIFFERENT INJECTION INTERVALS (SEC BETWEEN INJECTED VEH PER ROUTE).

Interval	75	60	45	30	15
AckPBSM	1.443	1.672	2.139	2.161	2.109
PBSM	3.012	3.662	5.518	7.194	11.084
DV-CAST	147.5	468.5	313.6	1037.5	953.2

and the message has been propagated backwards up to A (after applying the broadcast suppression technique). All vehicles are in idle state except A, which has the forwarding responsibility at that moment. Then, A speeds up and overtakes the remaining vehicles, forwarding the message to them and going to idle state. However, according to DV-CAST [7], the receivers discard the message as duplicated and the message custody is lost. No one will forward the message again even when new vehicles G, H emerge. This problem is derived from the different states in which DV-CAST operates depending on the traffic regime which is sensed by a vehicle, since it is hard to foresee every possible combination of movements in vehicular setups. AckPBSM and PBSM do not suffer from this problem.

Along low reliability of DV-CAST, the number of broadcast messages issued by the protocol is also low (see Figure 3). We observe that AckPBSM effectively reduces the number of transmissions per transmitting node with respect to PBSM. In addition, it remains almost constant regardless the traffic injection rate. This corroborates that AckPBSM takes advantage of the particular vehicular mobility patterns to improve the protocol efficiency.

B. Urban scenario

To simulate a section of an urban scenario, we have employed an square grid of 2 km per side. Each street has two lanes in opposite directions. Vehicles must stop at intersections when others are crossing, so that traffic jams are longer here than in the highway setup. DV-CAST has been included in this set of simulations just as a reference. Please recall that it is not designed for urban scenarios with intersections.

As expected, Table II shows again that AckPBSM and PBSM achieve the highest reliability. Furthermore, AckPBSM is the winner again with respect to the protocol efficiency (Table III). It reduces the number of transmissions per node up to a factor of 5.25 when compared to PBSM.

V. CONCLUSIONS

In this paper we have presented a broadcast protocol for vehicular networks which is able to achieve high reliability while minimizing the number of retransmissions. We have assessed its performance by means of a simulation-based study, where it has been shown to greatly outperform competing algorithms in a variety of vehicular scenarios and traffic conditions (from sparse to dense networks). To the best of our knowledge, this is the first broadcast protocol in the literature which has shown its suitability as an integral solution for VANET. Our future work will address the degree of compatibility of proposed protocol with DSRC.

ACKNOWLEDGMENT

This article has been partially funded by NSERC Canada, RSWRMA UK and the MARTA project (CENIT-2007-2008).

REFERENCES

[1] M. L. Sichitiu and M. Kihl, Inter-vehicle communication systems: a survey, *IEEE Communications Surveys & Tutorials*, Vol. 10, Issue 2, pp. 88–105, 2008.

[2] M. Sun, W. Feng, T. Lai, K. Yamada, H. Okada and K. Fujimura. Gps-based message broadcast for adaptive inter-vehicle communications, *IEEE Vehicular Technology Conference*, 52nd, Vol. 6, pp. 2685–2692, 2000.

[3] S. Biswas, R. Tatchikou and F. Dion, Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety, *IEEE Communications Magazine*, Vol. 44, Issue 1, pp. 74–82, Jan. 2006.

[4] R. Fracchia and M. Meo, Analysis and design of warning delivery service in inter-vehicular networks, *IEEE Transactions on Mobile Computing*, Vol. 7, No. 7, pp. 832–845, 2008.

[5] G. Korkmaz, E. Ekici, F. Ozguner and U. Ozguner, Urban multi-hop broadcast protocol for inter-vehicle communication systems, *First ACM Workshop on Vehicular Ad Hoc Networks VANET*, pp. 76–85, Oct. 2004.

[6] G. Korkmaz, E. Ekici and F. Ozguner, An efficient fully ad-hoc multi-hop broadcast protocol for inter-vehicular communication systems, *IEEE International Conference on Communications (ICC'06)*, Jun. 2006.

[7] O. K. Tonguz, N. Wisitpongphan, F. Bai, P. Mudalige and V. Sadekar, Broadcasting in VANET, *IEEE INFOCOM MOVE Workshop 2007*, Anchorage, AK, USA, May 2007.

[8] N. Wisitpongphan, O. K. Tonguz, J. S. Parikh, P. Mudalige, F. Bai and V. Sadekar, Broadcast storm mitigation techniques in vehicuar ad hoc wireless networks, *IEEE Wireless Communications Magazine*, pp. 84–94, Dec. 2007.

[9] I. Stojmenovic, M. Seddigh and J. Zunic, Dominating sets and neighbor elimination based broadcasting algorithms in wireless networks, *IEEE Transactions on Parallel and Distributed Systems*, Vol. 13, No. 1, pp. 14–25, Jan. 2002; *IEEE Transactions on Parallel and Distributed Systems*, Vol. 15, No. 11, pp. 1054–1055, Nov. 2004.

[10] A. A. Khan, I. Stojmenovic and N. Zaguia, Parameterless broadcasting in static to highly mobile wireless ad hoc, sensor and actuator networks, *22nd IEEE Int. Conf. on Advanced Information Networking and Applications (AINA2008)*, Ginowan, Okinawa, Japan, Mar. 2008.

[11] B. N. Clark, C. J. Colbourn and D. S. Johnson, Unit disk graphs, *Discrete Mathematics*, Vol. 86, Issue 1-3, pp. 165–177, 1991.