A Distributed Broadcast Algorithm for Ad Hoc Networks

Li Layuan, Li Chunlin, Sun Qiang

School of Computer Science, Wuhan University of Technology, Wuhan, 430063, P. R. China

Abstract. In mobile ad hoc networks, many unicast and multicast protocols depend on broadcast mechanism to finish control and route establishment functionality. In a straightforward broadcast by flooding, each node will retransmit a message to all it neighbors until the message has been propagated to the entire network. So it will become very inefficient and will be easy to result the broadcast storm problem. Thus an efficient broadcast algorithm should be used to less the broadcast storm caused by broadcast. Due to the dynamic nature of ad hoc networks, global information of the network is difficult to obtain, so the algorithm should be distributed. In this paper, an efficient distributed heuristic-based algorithm is presented. The algorithm is based on joint distance-counter threshold scheme. It runs in a distributed manner by each node in the network without needing any global information. Each node in an ad hoc network hears the message from its neighbors and decides whether to retransmit or not according to the signal strength and the number of the receiving messages. By using the JDCT algorithm, it's easy to find the nodes that consist of the vertices of the hexagonal lattice to cover the whole networks. The algorithm is very simple and it is easy to operate and has a good performance in mobile wireless communication environments. A comparison among several existing algorithms is conducted. Simulation results show that the new algorithm is efficient and robust.

1 Introduction

In mobile ad hoc networks, the research of routing is still at the beginning and some routing protocols have been put forward [1-10]. Most of these protocols depend on a broadcast mechanism [9-10]. In flooding, each node that receives a message retransmits that message to all it neighbors until the message has been propagated to the entire network. Despite its poor scalability and inefficient bandwidth usage [9], flooding is very useful because it can achieve maximal coverage, distance preservation, and redundancy. Without any reliance on knowledge about the network topology, flooding performance does not degrade with increased node mobility [10].

In this paper, we assume that mobile nodes in the mobile ad hoc networks share a single common channel and global network topology information is unavailable for each node. So each node of such a network can only communicate by broadcast. In order to less the broadcast storm [9], an efficient broadcast algorithm based on joint distance-counter threshold is proposed, which can run without any neighborhood

Layuan L., Chunlin L. and Qiang S. (2006).
A Distributed Broadcast Algorithm for Ad Hoc Networks.
In *Proceedings of the 5th International Workshop on Wireless Information Systems*, pages 86-93

information. This drastically reduces the effect of the mobility and no exchanged messages and control messages are needed. The goal of joint distance-counter threshold is to provide both a satisfied coverage and less broadcast and average latency, which a high coverage is guaranteed by distance threshold and a high saved broadcast and less average latency are guaranteed by counter threshold.

2 Notations and Assumptions

The symbols and definitions used in this paper are defined as follows:

Definition 1: The distance $d(x,y)$ between two nodes x and y in $G(t)$ is defined the distance between x and y within their transmission range r, $d(x,y) \le r$.

Definition 2: $c(x)$ denotes the number of received messages in node x during broadcast.

Definition 3: D_{Th} denotes a distance threshold, where $0 \le D_{Th} < r$.

Definition 4: C_{Th} denotes a counter threshold, where $C_{Th} \ge 0$.

Definition 5: $N(x)$ is a set of neighbors of node x, where for each neighbor y, $d(x,y) \le r$. For example, in Fig.2, the neighbors of node 1 consist of node 2, 3 and they can represent as $N(1) = \{2,3\}$.

Definition 6: $I(x)$ is a subset of $N(x)$, $I(x) \subseteq N(x)$ where for each of its member y, $y \in I(x)$ and $d(x,y) \le D_{Th}$. In Fig. 2, $I(1) = \{3\}$.

Definition 7: $E(x)$ is a subset of $N(x)$, $E(x) \subseteq N(x)$ where for each of its member y, $y \in E(x)$ and $D_{Th} < d(x,y) \le r$. For instance, in Fig. 2, $E(1) = \{2\}$.

Definition 8: $Rt(S)$ is a set of nodes that retransmit the message from source node S, where

$$Rt(S) = \left\{x \mid x \in \left\{\{E(x_1) \cap E(x_2) \cap \cdots E(x_k)\} \cup \{x \notin \{E(x_1) \cap E(x_2) \cap \cdots E(x_k)\}, c(x) < C_{TH}\}\right\}\right\} \cdot$$

$$x_1, x_2, \cdots, x_k \in N(x)$$

3 Joint Distance-Counter Threshold Broadcast Algorithm

In this section, an efficient distributed heuristic-based algorithm is presented. The goal of the proposed algorithm is to reduce the number of rebroadcasts without consume much resources of the network, such as bandwidth and energy.

3.1 Details of the Algorithm

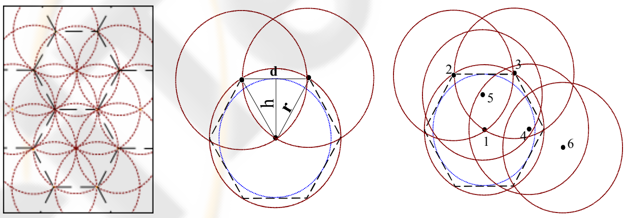
When a transmission of a broadcast message M by the source node occurred, all its neighbors will receive M in the same time. In order to alleviate the broadcast storm problem [9], a node has to assess the redundancy of a broadcast and decide whether to rebroadcast or not. The redundancy of a broadcast associates with the additional coverage caused by the source node and all its neighbors. [9] has revealed the relationship between the redundancy of a broadcast and the additional coverage. The further the distance between the source node and its neighbors is, and the larger additional coverage can be acquired. Moreover, the more neighbors of a node are, the less additional coverage is acquired.

Based on these relationships, joint distance-counter threshold broadcast algorithm is proposed in this paper. In this algorithm, according to the distance of its neighbors and the number of the received message, each node decides whether to retransmitting M or not. First, when a node x send a broadcast message M, all its neighbors will hear M and compute $d(x,y)$ according the receiving signal strength [24]. If $d(x,y_i) > D_{Th}, y_i \in N(x)$ then wait for a short time which is determined by a distance and counter relevant function. The delaying helps to avoid many nodes to transmit all at once. If node y_i hasn't received any messages during this short delay, it will transmit M at once. Otherwise; it will compute the distance from the sending node again. If $d(x,y_i) \le D_{Th}, y_i \in N(x)$ then depend on the number of the received message in node y_i to decide whether to retransmitting M or not. And if $c(y_i) < C_{Th}$, node y_i will wait a short delay and if there is not any other messages heard in this period, it will transmit M immediately. If $d(x,y_i) \ge D_{Th}$ then wait for a short time again.

3.2 Analysis

In flooding, each node has to retransmit the broadcast message. However, if a node is "covered" by other nodes, it needn't retransmit ever. The goal of an efficient broadcast algorithm has to find the covered nodes and let those "uncovered" nodes retransmit the message.

In [25], Kershner has pointed out that the hexagonal lattice is the most efficient arrangement to cover the plane (Fig.1. a). If the vertices of such hexagonal lattice are consisted by nodes of ad hoc networks, the broadcast message transmitted only by these nodes can reach all nodes in a network.



(a) (b) (c)

Fig. 1. The coverage problem.

Lemma 1: The JDCT algorithm can find the nodes that consist of the vertices of the hexagonal lattice if these nodes exist.

Proof. As we know, in a regular polygon that each side is *d* (Fig.1. b), we can get

$$d = 2r\sin\left(\frac{\pi}{N}\right) \quad (1)$$

$$h = r\cos\left(\frac{\pi}{N}\right) \quad (2)$$

where N is the vertices number of the regular polygon. As $d \approx 0.87r, h \approx 0.9r$ when $N = 7$, the distance between two nodes belonging to $E(x)$ is larger than $0.87r$ when $0.87r < D_{Th} \le r$. And as $d \approx 1.17r, h \approx 0.8r$ when $N = 5$, the JDCT algorithm can always find the nodes that consist of the vertices of the hexagonal lattice if these nodes exist when $0.87r < D_{Th} \le r$.□

Lemma 2: The JDCT algorithm can work well both in even and uneven distributed ad hoc networks.

Proof. In an even distributed ad hoc network, as using the distance threshold, the algorithm has more chance to find the nodes that consist of the vertices of the hexagonal lattice and assures the high reachability. When there cannot find any nodes that consist of the vertices of the hexagonal lattice i.e. there is no node belong to $E(x)$ such as the nodes in an uneven distributed ad hoc networks or the satisfied nodes number is less than six, the counter threshold C_{Th} is use to guarantee a high reachability in the JDCT algorithm. In this case, only those whose number of received messages is less than C_{Th} can retransmit the message. So the JDCT algorithm can work well both in even and uneven distributed ad hoc networks. Fig.4. c gives an example, where $C_{Th} = 3$, node 4 will retransmit the message.□

Lemma 3: In JDCT algorithm, each node can determine whether or not to rebroadcast in no more than $O(k)$ computation time ($k = |E(x)|$).

Proof. According to definition 8, a node can retransmit the message when it belongs to $Rt(S)$. The computation time in each node x is determined by the time consumed to estimate whether $x \in \{\{E(x_1) \cap E(x_2) \cap \cdots E(x_k)\} \cup \{x \notin \{E(x_1) \cap E(x_2) \cap \cdots E(x_k)\}, c(x) < C_{TH}\}\}, N(x) = \{x_1, x_2, \cdots, x_k\}$ or not. The overall worst-case time complexity of the JDCT algorithm occurs when a node belongs to all its retransmitting neighbors' own $E(x)$ where $|E(x)| = k$. So in JDCT algorithm, each node can determine whether or not to rebroadcast in no more than $O(k)$ computation time. □

Lemma 4: The JDCT algorithm is distributed, parameter-less and less communication overhead.

Proof. Since the JDCT algorithm is running in each node and each node only depends on the message it has heard from its neighbors to decide whether to retransmit or not, the JDCT algorithm is distributed, parameter-less and less communication overhead. □

Theorem 1: The JDCT algorithm satisfies the properties of the efficient distributed broadcast algorithm in the ad hoc networks.

Proof. According to lemma 1 and lemma 2, through adjust the value of D_{Th} and C_{Th}, the algorithm gets a high rebroadcast saving both in even and uneven distributed ad hoc networks. According to lemma 3, each node decides whether to retransmit or not only with limited steps. And according to lemma 4, the algorithm is distributed, parameter-less and less communication overhead. Therefore, the JDCT algorithm satisfies the properties of the efficient distributed broadcast algorithm. □

4 Simulation Results

To evaluate the new broadcast algorithm and compare it to existing algorithms, simulations are performed. A Mobility Framework for OMNeT++ (a discrete event simulator written in C++, described in [25]) is used as a tool. The size of the network is 100 nodes in a 1000*1000 meter square. The nodes are uniformly distributed all over the region. Nodes in the simulation move according to “random waypoint” model [27]. The mobility speed of a node is set from 0m/s to 30m/s. The CSMA/CA is used as the MAC layer in our experiments. The transmitting radius of each node is about 231 meters and channel capacity is 10Kbits/sec. Four distributed broadcast algorithms are chose to be compared and they are listed below.

– **SB:** straightforward broadcast algorithm
– **DB:** distance-based broadcast algorithm
– **CB:** counter-based broadcast algorithm
– **JDCT:** JDCT broadcast algorithm

The performance measures of interest are:

• **Average latency:** defined as the interval between its arrival and the moment when either all nodes have received it or no node can rebroadcast it further.

• **Ratio of Saved Rebroadcast (RSR):** The total number of nodes not delivered broadcast packets is divided by the total number of nodes receiving the broadcast message.

• **Ratio of Collision (RC):** The total number of collision is divided by the total number of packets supposed to be delivered during broadcast.

• **Total Number of Contention (TNC):** the total number of contention during broadcast.

• **Total Number of Received Messages (TNRM):** the total number of received messages is the sum of the number of messages heard by each node during broadcast.

The first set of experimental results (Fig.2) demonstrates average RSR versus distance threshold using distance-based algorithm and JDCT algorithm with R=1. The result shows that JDCT algorithm obtains higher SRS than the distance-based algorithm. When $D_{Th} \approx 0.9r = 210\text{meters}$, the RSR of both distance-based algorithm and JDCT algorithm get their maximum value of SRS, about 52% in JDCT and 28% in DB. This is understandable, because with the increase in threshold value, number of retransmitting nodes decrease. When $D_{Th} > 0.9r$, the distance-based algorithm can’t cover the whole network in our simulation scenarios, however, the JDCT can cover the whole network. When $0.87r < D_{Th} \le r$, the JDCT can find nodes to consist of the vertices of the hexagonal lattice. Thus a high RSR value can be acquired. As the threshold value increases, there are not enough nodes to be found by DB to cover the whole network. However, although nodes selected by the distance threshold are not enough to consist of the vertices of the hexagonal lattice, as the counter threshold is used, the JDCT can also keep a full coverage to the network.

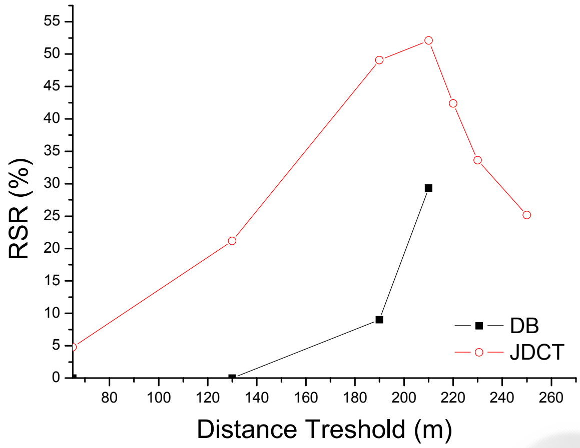


Fig. 2. Distance threshold D_{Th} vs. RSR (R=1).

In Fig.3-5, the results are gotten with the parameters of $R = 1, D_{Th} = 0.9r, C_{Th} = 3$. Fig.6 shows the ratio of saved rebroadcast using different algorithms with varying node speeds (from 0 to 30m/sec). The average latency of different broadcast algorithms with varying node speeds is reported in Fig.4. Fig.5 gives the ratio of collision in different algorithms with varying node speeds.

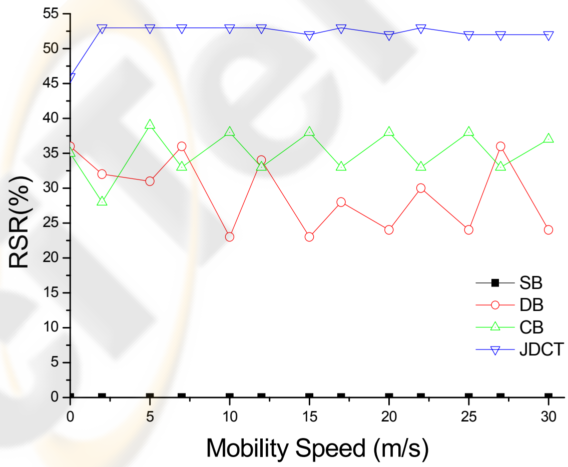


Fig. 3. Ratio of Saved Rebroadcast vs. Mobility speed.

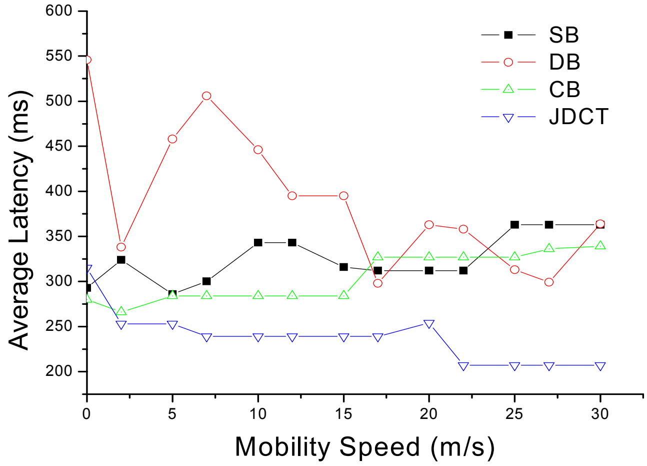


Fig. 4. Average Latency vs. Mobility speed Ratio of Saved Rebroadcast vs. Mobility speed.

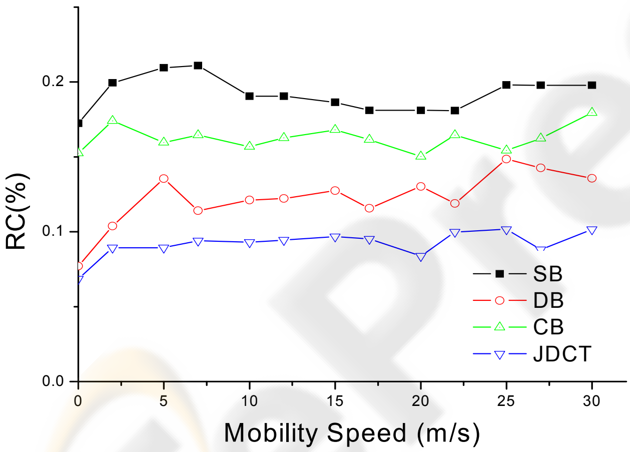


Fig. 5. Ratio of Collisions vs. Mobility speed.

5 Conclusion

Building efficient broadcast algorithm for ad hoc networks is challenging due to the dynamic of the nodes. In ad hoc networks, centralized algorithms are not suitable because the global information is impossible to get. And as ad hoc network is a multihop networks, there always exists hidden/exposed terminals. Many broadcast algorithms depended on local information such as k-hop neighbor information will not get a desirable performance when local information cannot be acquired correctly. As ad hoc networks are resource-limited networks, algorithms based on exchanging control messages such as *hello* message are also unsuitable. And some algorithms used GPRS are constrained by nodes' limited energy.

In this paper, an efficient distributed heuristic-based algorithm named JDCT algorithm is presented. The algorithm is based on joint distance and counter threshold scheme. It runs in a distributed manner by each node in the network without needing any global information. The experiments have demonstrated the efficiency of proposed broadcast algorithm. The broadcast storm problem is alleviated by significant reduction in the number of rebroadcast nodes, contention and collision in the network. Its efficiency and robustness in mobile networks make it a good choice for mobile ad hoc networks. Our future work includes a performance evaluation of the JDCT broadcast algorithm in realistic simulation environments with packet collision and node mobility. In addition, we will embed our JDCT algorithm to some routing protocols such as AODV protocol to investigate its efficiency in ad hoc networks.

Acknowledgments

This work is proudly supported in part by the Grand Research Problem of the National Natural Science Foundation of China under Grant No. 90304018 and Wuhan key project.

References

1. Li Layuan, Li Chunlin. A QoS multicast routing protocol for dynamic group topology [J]. Inf. Sci., 2005,169(1-2): 113-130.
2. S.Ramanathan, Martha Steenstrup. A survey of routing techniques for mobile communications networks [J]. Mobile Networks and Applications, 1996, 1(2): 89-104.
3. C. E. Perkins, E. M. Royer. Ad hoc on-demand distance vector (AODV) routing[C]. In Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, Feb. 1999.
4. Li Layuan,Li Chunlin. A routing protocol for dynamic and large computer networks with clustering topology [J]. Computer Communications, 2000, Elsevier, UK, 23(2): 171-176.
5. Sung-Ju Lee, William Su, Mario Gerla. Ad hoc Wireless Multicast with Mobility Prediction[C]. IEEE ICCCN'99, Boston, MA, Oct. 1999.
6. David Johnson, David Maltz. Dynamic source routing in ad hoc wireless networks [J]. Mobile Computing, Boston: Kluwer Academic Publishers, 1996
7. Li Layuan, Li Chunlin. A distributed QoS-aware multicast routing protocol [J]. Acta Informatica, Springer, Germany, 2003, 40 (3): 221-233.
8. J. Cartigny and D. Simplot, Border Node Retransmission Based Probabilistic Broadcast Protocols in Ad-Hoc Networks [C]. In Proc. 36th International Hawaii International Conference on System Sciences (HICSS'03), Hawaii, USA. 2003.
9. Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu. The Broadcast Storm Problem in a Mobile Ad Hoc Network [J]. Wireless Networks, 2002, 5 (8): 153-167.
10. E. Royer and C-K. Toh. A Review of Current Routing Protocols for Ad-Hoc Mobile Wireless Networks [J]. IEEE Personal Communications Magazine, 1999,4: 46-55.