www.ijiarec.com

ISSN:2348-2079

Volume-5 Issue-2

International Journal of Intellectual Advancements and Research in Engineering Computations

Measurement based roadside content delivery system

[1]Mrs.K.E.Eswari M.C.A., ME., Associate Professor,
[2]Mr.M.V.Bibin Final M.C.A,
Department of MCA, Nandha Engineering College (Autonomous), Erode-52

E-mail id: bibinmv09@gmail.com

Abstract*—This paper presents two efficient broadcasting algorithms based on 1-hop neighbor information. In the first part of the paper, we consider sender-based broadcasting algorithms, specifically the algorithm proposed by Liu et al. In their paper, Liu et al. proposed a sender-based broadcasting algorithm that can achieve local optimality by selecting the minimum number of forwarding nodes in the lowest computational time complexity O(n log n), where n is the number of neighbors. We show that this optimality only holds for a subclass of sender-based algorithms. We propose an efficient sender-based broadcasting algorithm based on 1-hop neighbor information that reduces the time complexity of computing forwarding nodes to O(n). In Liu et al.'s algorithm, n nodes are selected to forward the message in the worst case, whereas in our proposed algorithm, the number of forwarding nodes in the worst case is 11. In the second part of the paper, we propose a simple and highly efficient receiver-based broadcasting algorithm. When nodes are uniformly distributed, we prove that the probability of two neighbor nodes broadcasting the same message exponentially decreases when the distance between them decreases or when the node density increases. Using simulation, we confirm these results and show that the number of broadcasts in our proposed receiver-based broadcasting algorithm can be even less than one of the best known approximations for the minimum number of required broadcasts.

***Index Terms*—Wireless ad hoc networks, flooding, broadcasting, localized algorithms.**

1 INTRODUCTION

Broadcasting is a fundamental communication operation in which one node sends a message to all other nodes in the network. Broadcasting is widely used as a basic mechanism in many ad hoc network protocols. For example, ad hoc on-demand routing protocols such as AODV and DSR typically use broadcasting in their route discovery phase. Broadcasting is also used for topology updates, for network maintenance, or simply for sending a control or warning message. The simplest broadcasting algorithm is flooding, in which every node broadcasts the message when it receives it for the first time. Using flooding, each node receives the message from all its neighbors in a collision-free network. Therefore, the broad-cast redundancy significantly increases as the average number of neighbors increases. High broadcast redundancy can result in high power and bandwidth consumption in the network. Moreover, it increases packet collisions, which can lead to additional transmissions. This can cause severe network congestion or significant performance degradation, a phenomenon called the broadcast storm problem. Consequently, it is crucial to design efficient broadcasting algorithms to reduce the number of required transmissions in the network.

A set of nodes is called a Dominating Set (DS) if any node in the network either belongs to the set or is a 1-hop neighbor of a node in the set. The set of broadcasting nodes forms a Connected DS (CDS). Therefore, the minimum number of required broadcasts is not less than the size of the minimum CDS. Unfortunately, finding the minimum CDS is NP-hard, even for the unit disk graphs. However, there are some distributed algorithms that can find a CDS whose size is smaller than a constant factor of the size of the minimum CDS. These algorithms can be employed to find a small-sized CDS that can be used as a virtual backbone for broadcasting in ad hoc networks. However, this approach is not efficient in networks with frequent topology changes, as maintaining a CDS is often costly.

The main objective of efficient broadcasting algorithms is to reduce the number of broadcasts while keeping the bandwidth and computational overhead as low as possible. One approach to classify broadcasting algorithms is based on the neighbor information they use. Some broadcasting algorithms such as flooding and probabilistic broadcasting algorithms do not rely on neighborhood knowl-edge. These algorithms cannot typically guarantee full delivery and/or effectively reduce the number of broad-casts. Moreover, to decide whether or not to broadcast, they may use a threshold (such as probability of broadcast), which may not be easy to find for different network situations. In the second category, broadcasting algorithms require having 2-hop or more neighbor information. The broadcasting algorithms in this category can reduce the number of broadcasts in the network and guarantee full delivery. However, they may induce high overhead in highly dynamic networks as they need to maintain 2-hop network connectivity.

In this paper, we propose two broadcasting algorithms based on 1-hop neighbor information. The first proposed Algorithm is a sender-based algorithm. In sender-based algorithms, the broadcasting nodes select a subset of their neighbors to forward the message. We compare our proposed broadcasting algorithm to one of the best sender-based broadcasting algorithms that use 1-hop information. In , Liu et al. propose a broadcasting algorithm that reduces the number of broadcasts and achieves local optimality by selecting the minimum number of forwarding nodes with minimum time complexity O(n log n), where n is the number of neighbors. We show that this optimality only holds for a subclass of sender-based broadcasting algorithms employing 1-hop information and prove that our proposed sender-based algorithm can achieve full delivery with time complexity O(n). Moreover, Liu et al.'s algorithm selects n forwarding nodes in the worst case, while our proposed algorithm selects 11 nodes in the worst case. Based on our simulation results, our sender-based algorithm results in fewer broadcasts than does Liu et al.'s algorithm. All these interesting properties are achieved at the cost of a slight increase in end-to-end delay. Thus, our first proposed algorithm is preferred to Liu et al.'s algorithm when the value of n is typically large, and it is important to bound the packet size.

We also propose a receiver-based broadcasting algorithm in this paper. In receiver-based algorithms, the receiver decides whether or not to broadcast the message. The proposed receiver-based algorithm is a novel broadcasting algorithm that can significantly reduce the number of broadcasts in the network. We show that using our proposed receiver-based algorithm, two close neighbors are not likely to broadcast the same message. In other words, we prove that the probability of broadcast for a node NA exponentially decreases when the distance between NA and its broadcasting neighbor decreases or when the density of nodes increases. Based on our experimental results, the number of broadcasts using our receiver-based algorithm is less than one of the best known approximations for the minimum number of required broadcasts.

2 SYSTEM MODEL

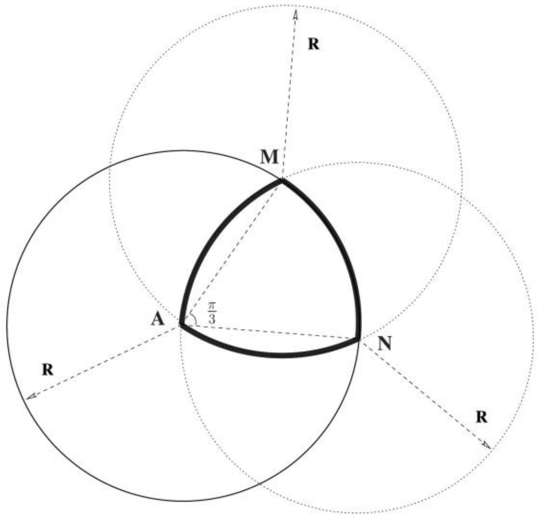
Our system model is very similar to that used by Liu et al. We assume that all nodes are located in a 2D plane and have a transmission range of R. Therefore, the topology of the network can be represented by a unit disk graph. We assume that the network is connected. Two nodes are considered neighbors if they are in the transmission range of each other. We suppose that each node knows its location via a localization technique such as Global Positioning System (GPS) or the lightweight techniques. Each node periodically broadcasts a very short Hello message, which includes its ID and position. Thus, each node gets the position of its neighbors as well. In the medium access control (MAC) layer, we assume that scheduling is done according to the p-persistent CSMA/CA protocol, which is based on IEEE 802.11 in the broadcast mode. In the p-persistent CSMA/CA protocol, when a node has a message to transmit, it initiates a defer timer by a random number and starts listening to the channel. If the channel is busy, it continues to listen until the channel becomes idle. When the channel is idle, it starts decrementing the defer timer at the end of each time unit. The message is broadcast when the timer expires.

3 AN EFFICIENT SENDER-BASED BROADCASTING ALGORITHM

3.1 Algorithm Structure

Our first proposed broadcasting algorithm is a sender-based algorithm, i.e., each sender selects a subset of nodes to forward the message. Each message can be identified by its source ID and a sequence number incremented for each message at the source node. Algorithm 1 is a general sender-based broadcasting algorithm and indicates the structure of our proposed sender-based broadcasting algorithm. Upon expiration of the timer, the algorithm requests the MAC layer to schedule a broadcast. The message scheduled in the MAC layer is buffered and then broadcast with a probability p. This adds another delay in broadcasting the message. The MAC-layer delay in IEEE 802.11 is a function of several factors including the network traffic. Note that there is a chance that a node changes its decision during the MAC-layer delay due to receiving other copies of the message. This chance is not negligible when the delay in the MAC layer is comparable to the average value of the timer set in the broadcasting algorithm. As stated in , one solution to this problem is a cross-layer design in which the network layer is given the ability to modify or remove packets that are present in the MAC-layer queue. This solution allows the broadcasting algorithms to perform close to their ideal performance even for very small average timer values. In the entire paper, we assume that either the MAC-layer delay is negligible compared to the average delay set by the algorithm or the network layer (hence, the algorithm) is able to modify or remove packets buffered in the MAC-layer queue.

The sender-based broadcasting algorithms can be divided into two subclasses. In the first subclass, each node decides whether or not to broadcast solely based on the first received message and drops the rest of the same messages that it receives later. Liu et al.'s algorithm falls in this subclass and achieves local optimality by selecting the minimum number of forwarding nodes in the lowest computational time complexity.



A bulged slice around A.

In the second subclass of sender-based broadcasting algorithms, each node can decide whether or not to broadcast after each message reception. However, if a node broadcasts a message, it will drop the rest of the same messages that it receives in the future. Therefore, each message is broadcast once at most by a node using the broadcasting algorithms in both subclasses. Our first proposed broadcasting algorithm falls in this subclass of sender-based broadcasting algorithms. We show that the proposed algorithm can reduce both the computational complexity of selecting the forwarding nodes and the maximum number of selected nodes in the worst case.

Algorithm 1 shows the basic structure of our proposed sender-based broadcasting algorithm. As shown in Algorithm 1, each node schedules a broadcast for a received message if the node is selected by the sender and if it has not scheduled the same message before. Clearly, each message is broadcast once at most by a node, which is similar to Liu et al.'s algorithm. However, in Liu et al.'s algorithm, each node may only schedule a broadcast when it receives a message for the first time. In contrast, in Algorithm 1, a broadcast schedule can be set at any time. For example, a message can be dropped after the first reception but scheduled for broadcast the second time. Clearly, the main design issue in Algorithm 1 is how to select the forwarding nodes.

Algorithm 1. A general sender-based algorithm

```
Extract information from the received message M
if M has been scheduled for broadcast or does not Contain node's ID then
    drop the message
else
    set a defer timer
end if
When defer timer expires
    Select a subset of neighbors to forward the message
    Attach the list of forwarding node to the message
    Schedule a broadcast
```

3.2 Forwarding-Node Selection Algorithm

A node can have several different B-coverage sets. Therefore, there is more than one slice-based selection algorithm. For example, a trivial slice-based selection algorithm would be one that selects all of the neighbors as the B-coverage set. Clearly, this algorithm will result in flooding if it is used as the forwarding-node selection scheme in Algorithm 1. In this section, we first show that Algorithm 1 can achieve full delivery if it uses any slice-based algorithm to select the forwarding nodes. We then present an efficient slice-based algorithm that selects 11 nodes in the worst case and has computational complexity O(n), where n is the number of neighbors.

3.3 Reducing the Number of Forwarding Nodes

In the sender-based broadcasting algorithms, each broadcasting node attaches a list of its selected forwarding nodes to the message before broadcasting it. This procedure will increase the bandwidth and power required to broadcast the message. As shown earlier, our proposed slice-based selection algorithm reduces the number of selected forwarding nodes to 11 in the worst case. In this section, we show how to further reduce the number of selected nodes

3.4 Maximizing the Minimum Node Weight of B-Coverage Set

Suppose node NA assigns a weight to each of its neighbors. The weight can represent the neighbor's battery lifetime, its distance to NA, the average delay of the node, the level of trust, or a combination of them. In some scenarios, we may desire to find a B-coverage set such that its minimum node weight is the maximum or its maximum node weight is the minimum among that of all B-coverage sets. For example, assume that the weight of each node represents its battery lifetime in a wireless network. It may be desirable to select the nodes with a higher battery lifetime to forward the message in order to keep the nodes with a lower battery lifetime alive. Algorithm 3 shows how to find a B-coverage set such that its minimum node weight is the maximum among that of all B-coverage sets. A similar approach can be used to find a B-coverage set such that its maximum node weight is the minimum.

3.5 Similarity with a Topology Control Algorithm

The objective of the algorithm is to minimize the transmission power of each node without violating the network connectivity. In order to do that, each node NA transmits with the minimum power P_ such that in every nonempty cone of degree _ around NA, there is some node that NA can reach with power P_. A cone is nonempty if there is at least a node in the cone that NA can reach using its maximum power.

4 .A HIGHLY EFFICIENT RECEIVER-BASED BROADCASTING ALGORITHM

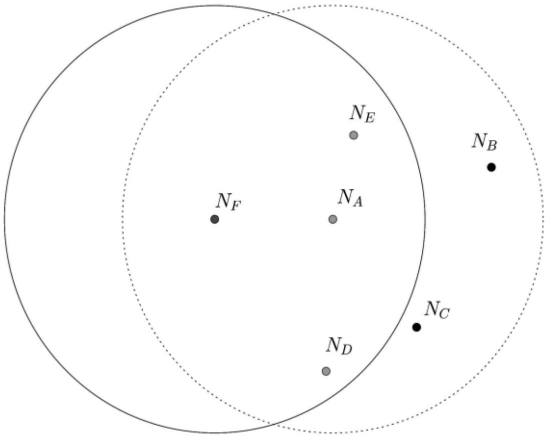
In this section, we propose a novel receiver-based broadcasting algorithm that can significantly reduce redundant broadcasts in the network. As mentioned earlier, in receiver-based broadcasting algorithms, the receiver of the message decides whether or not to broadcast the message. Therefore, a potential advantage of receiver-based broadcasting algorithms over sender-based ones is that they donor increase the size of the message by adding a list of forwarding nodes.

4.1 Algorithm Structure

Algorithm 4 shows a general approach used in several receiver-based broadcasting algorithms. Our proposed receiver-based broadcasting algorithm employs this approach. Clearly, the main design challenge of Algorithm 4 is to determine whether or not to broadcast a received message. A trivial algorithm is to refrain broad-casting if and only if all the neighbors have received the message during the defer period. Although this algorithm is simple to implement, it has limited effect in reducing the number of redundant broadcasts. Suppose N_A's defer time expires at t_0. Using the above strategy, node N_A will broadcast if some of its neighbors (at least one) have not received the message by t_0. However, this broadcast is redundant if all such neighbors receive the message from other nodes after time t_0. This scenario typically occurs when t_0 is small compared to the maximum defer time. In the next section, we introduce a responsibility-based scheme (RBS) that further reduces the redundant broadcasts without any changes in the MAC-layer defer-time design.

Algorithm 4. A general receiver-based algorithm

```
Extract information from the received message M
if M has been received before then
  drop the message
else
  set a defer timer
end if
When defer timer expires
  decide whether or not to schedule a broadcast
```



An example of an RBS decision.

4.2 Responsibility-Based Scheme

The main idea of this node avoids broadcasting if it is not responsible for any of its neighbors. A node N_A is not responsible for a neighbor N_B if N_B has received the message or if there is another neighbor N_C such that N_C has received the message and N_B is closer to N_C than it is to N_A. Suppose N_A stores IDs of all its neighbors that have broadcast the message during the defer period. When executed by a node N_A, Algorithm 5 first uses this information to determine which neighbors have not received the message. It then returns false if and only if it finds a neighbor N_B that has not received the message and for any N_A's neighbor N_C that has received the message. The output of RBS determines whether or not the broadcast is redundant.

4.3 A Property of the Proposed RBS

In the simulation section, we show that the proposed RBS can significantly reduce the number of broadcasts in the network. In particular, our simulation shows that using RBS, the average number of broadcasts is less than one of the best known approximations for the minimum number of required broadcasts. Relaxing Some System Model Assumptions We assumed in Section 2 that the nodes are placed in a 2D plane. Therefore, the proposed receiver-based algorithm can also achieve full delivery when the nodes are distributed in a 3D space. Note that in this case, RBS uses 3D node positions.

5 . SIMULATION

5.1 Average Number of Nodes Selected by the Proposed Sliced-Based Algorithm

To avoid the complexity of mathematical analysis, we used a simulation to find the average number of selected nodes. We randomly put n points inside a circle with radius R. We then ran the proposed selection algorithm and obtained the number of selected nodes. To get the average number of selected nodes, we ran simulation 106 times for each given n. the average number of selected nodes is less than six and approaches five when n increases. Note that the proposed sliced-based selection algorithm does not necessarily select a B-coverage with a minimum number of nodes. However, there is a sliced-based selection algorithm that can find a B-coverage with a minimum number of nodes in O(n log n) and can consequently reduce the average number of selected nodes.

5.2 Probability of Broadcast Using the Proposed RBS

Suppose that the proposed receiver-based algorithm is used for broadcasting in the network. Assume that node NB receives a message from NA for the first time. It has been proven that the probability of NB broadcasting the message exponentially decreases when the distance AB decreases or when the node density increases. We used simulation to confirm this theoretical result. For the simulation, we considered two nodes NA and NB with distance 0 < d <- R from each other. We uniformly placed nodes with density inside the network and checked whether or not NB was required to broadcast the message. We ran simulation 106 times for a given a and R. We then estimated by the ratio of the number of times NB was required to broadcast over the total number of runs.

6. CONCLUSION AND FUTURE WORK

In the first part of this paper, we proposed a forwarding-node selection algorithm that selects at most 11 nodes in O (n), where n is the number of neighbors. This limited number of nodes is an improvement over Liu et al.'s algorithm, which selects n nodes in the worst case and has time complexity O (n log n). Moreover, we showed that our proposed forwarding-node selection algorithm results in fewer broadcasts in the network. In the second part of the paper, we proposed an efficient receiver-based algorithm and showed why it significantly reduces the number of forwarding nodes in the network. We also relaxed some system model assumptions that are typically used in the broadcasting algorithms. Interestingly, the 2-

hop-based version of our proposed receiver-based algorithm can guarantee constant approximation to the optimal solution (minimum CDS). As far as the authors know, this is the first broadcasting algorithm that constructs a CDS "on the fly" and can guarantee both full delivery and a constant approximation ratio to the optimal solution. As part of our future work, we will investigate the necessary conditions to guarantee both full delivery and constant approx-Imation ratio to the minimum CDS

REFERENCES

[1] C. Perkins, Ad Hoc on Demand Distance Vector (AODV) Routing, IETF Internet draft, work in progress, 1997.

[2] D. Johnson and D. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," Mobile Computing, T. Imielinski and H.F. Korth, eds., Kluwer Academic Publishers, 1996.

[3] S. Ni, Y. Tseng, Y. Chen, and J. Sheu, "The Broadcast Storm Problem in a Mobile Ad Hoc Network," Proc. ACM MobiCom '99, pp. 151-162, 1999.

[4] M. Garey and D. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman, 1990.

[5] B. Clark, C. Colbourn, and D. Johnson, "Unit Disk Graphs," Discrete Math., vol. 86, pp. 165-177, 1990.

[6] P. Wan, K. Alzoubi, and O. Frieder, "Distributed Construction of Connected Dominating Set in Wireless Ad Hoc Networks," Proc. IEEE INFOCOM '02, vol. 3, pp. 1597-1604, 2002.

[7] S. Funke, A. Kesselman, U. Meyer, and M. Segal, "A Simple Improved Distributed Algorithm for Minimum CDS in Unit Disk Graphs," ACM Trans. Sensor Networks, vol. 2, no. 3, pp. 444-453, 2006.

[8] H. Liu, P. Wan, X. Jia, X. Liu, and F. Yao, "Efficient Flooding Scheme Based on 1-Hop Information in Mobile Ad Hoc Net-works," Proc. IEEE INFOCOM, 2006.

[9] Y. Tseng, S. Ni, and E. Shih, "Adaptive Approaches to Relieving Broadcast Storms in a Wireless Multihop Mobile Ad Hoc Networks," Proc. 21st Int'l Conf. Distributed Computing Systems (ICDCS '01), pp. 481-488, 2001.

[10] Y. Sasson, D. Cavin, and A. Schiper, "Probabilistic Broadcast for Flooding in Wireless Mobile Ad Hoc Networks," Proc. IEEE Wireless Comm. and Networking Conf. (WCNC '03), pp. 1124-1130, 2003.

[11] W. Lou and J. Wu, "Double-Covered Broadcast (DCB): A Simple Reliable Broadcast Algorithm in Manets," Proc. IEEE INFOCOM '04, pp. 2084-2095, 2004.

[12] J. Wu and F. Dai, "Broadcasting in Ad Hoc Networks Based on Self-Pruning," Proc. IEEE INFOCOM '03, pp. 2240-2250, 2003.

[13] W. Peng and X. Lu, "On the Reduction of Broadcast Redun-dancy in Mobile Ad Hoc Networks," Proc. ACM MobiHoc '00, 129-130, 2000.

[14] T. He, C. Huang, B. Blum, J. Stankovic, and T. Abdelzaher, "Range-Free Localization Schemes in Large Scale Sensor Net-works," Proc. ACM MobiCom '03, pp. 81-95, 2003.

[15] A. Keshavarz-Haddad, V. Ribeiro, and R. Riedi, "DRB and DCCB: Efficient and Robust Dynamic Broadcast for Ad Hoc and Sensor Networks," Proc. Fourth Ann. IEEE Conf. Sensor, Mesh and Ad Hoc Comm. and Networks (SECON '07), June 2007.

[16] R. Wattenhofer, L. Li, P. Bahl, and Y. Wang, "Distributed Topology Control for Power Efficient Operation in Multihop Wireless Ad Hoc Networks," Proc. IEEE INFOCOM '01, pp. 1388-1397, 2001.