On Reducing Broadcast Redundancy in Ad Hoc Wireless Networks

Wei Lou, *Student Member, IEEE*, and Jie Wu, *Senior Member, IEEE*

Abstract—Unlike in a wired network, a packet transmitted by a node in an ad hoc wireless network can reach all neighbors. Therefore, the total number of transmissions (forward nodes) is generally used as the cost criterion for broadcasting. The problem of finding the minimum number of forward nodes is NP-complete. Among various approximation approaches, dominant pruning [7] utilizes 2-hop neighborhood information to reduce redundant transmissions. In this paper, we analyze some deficiencies of the dominant pruning algorithm and propose two better approximation algorithms: total dominant pruning and partial dominant pruning. Both algorithms utilize 2-hop neighborhood information more effectively to reduce redundant transmissions. Simulation results of applying these two algorithms show performance improvements compared with the original dominant pruning. In addition, two termination criteria are discussed and compared through simulation under both the static and dynamic environments.

Index Terms—Ad hoc wireless networks, broadcast, dominant pruning, flooding.

1 Introduction

In areas where there is little or no communication infrastructure or the existing infrastructure is inconvenient to use, wireless mobile users may still be able to communicate through the formation of an *ad hoc wireless network*. An ad hoc wireless network is a collection of wireless mobile hosts forming a temporary network without the aid of any centralized administration or standard support services [15]. In such a network, each mobile node operates not only as a host but also as a router. The applications of ad hoc wireless networks range from military use in battlefields, personnel coordinate tools in emergency disaster relief, to interactive conferences that temporarily formed using PDAs.

Broadcasting to all nodes in a network has extensive applications in ad hoc wireless networks, such as when used in the route query process in several routing protocols [6], [11], [13], when sending an error message to erase invalid routes [10], or when used as an efficient mechanism for reliable multicast in fast moving ad hoc wireless networks [5]. The way that packets are transmitted in ad hoc wireless networks is quite different than the way that those are transmitted in wired networks; the significant difference is that, when a host sends a packet, all of its neighbors will receive that packet (i.e., each node operates under the *promiscuous receive mode*). Therefore, the total number of transmissions (forward nodes) is generally used as the cost criterion for broadcasting. Basically, source and forward nodes form a *flood tree* such that any other node in the network is adjacent to a node in the tree. The problem of finding a minimum flood tree that has the minimum number of forward nodes is proven to be NP-complete [7]. Even when a minimum flood tree is identified, maintaining such a tree in a mobile environment is too costly to be useful in practice.

A straightforward approach for broadcasting is *blind flooding*, in which each node will be obligated to rebroadcast the packet whenever it receives the packet for the first time. Blind flooding will generate many redundant transmissions. Fig. 1 shows a network with three nodes. When node u broadcasts a packet, both nodes v and w receive the packet. Then, v and w will rebroadcast the packet to each other. Apparently, the last two transmissions are unnecessary. Redundant transmissions may cause a more serious *broadcast storm problem* [9] in which redundant packets cause contention and collision.

Many broadcast algorithms besides blind flooding have been proposed [1], [2], [7], [9], [12], [14], [16]. These algorithms utilize neighborhood and/or history information to reduce redundant packets. The *dominating pruning* (DP) algorithm [7] is one of the promising approaches that utilizes 2-hop neighborhood information to reduce redundant transmissions. The DP algorithm can also be considered as an approximation to the minimum flood tree problem.

In this paper, we point out some deficiencies of the DP algorithm, which does not eliminate all redundant transmissions based on 2-hop neighborhood information. Two algorithms, *total dominant pruning* (TDP) and *partial dominant pruning* (PDP), are proposed. Both algorithms utilize neighborhood information more effectively. Simulation results of applying these two algorithms show performance improvements compared with the original dominant pruning. In addition, two termination criteria are discussed and compared through simulation under both the static and dynamic environments.

The rest of the paper is organized as follows: Section 2 discusses some related work on reducing broadcast redundancy. Section 3 gives a graph model for ad hoc

• *The authors are with the Department of Computer Science and Engineering, Florida Atlantic University, Boca Raton, FL 33431. E-mail: {wlou, jie}@cse.fau.edu.*

Manuscript received 18 Oct. 2001; revised 27 June 2002; accepted 17 July 2002.
For information on obtaining reprints of this article, please send e-mail to: tmc@computer.org, and reference IEEECS Log Number 12-102001.

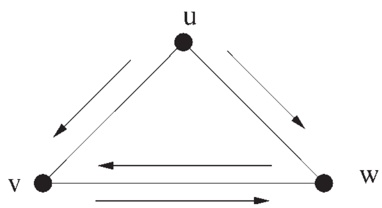


Fig. 1. Redundant transmissions by blind flooding.

wireless networks. Details about the DP algorithm are also presented. Two proposed broadcast algorithms are given in Section 4, with an example. In Section 5, we discuss two termination criteria for the broadcast process. Simulation results are shown in Section 6. Finally, Section 7 concludes the paper and outlines one future work.

2 Related Work

Efficient broadcasting in ad hoc wireless networks has been extensively studied in [1], [2], [7], [9], [12], [14], [16]. In [7], Lim and Kim prove that building a minimum flooding tree is the same as finding a *minimum connected dominating set (MCDS)* in a network, which is an NP-complete problem. A subset of nodes is called a *dominating set* if every node in the network is either in the set or a neighbor of a node in the set. They also provide two approximation algorithms: self-pruning and dominant pruning. The self-pruning algorithm exploits the knowledge of directly connected neighborhood information only. A node does not need to rebroadcast a packet if all its neighbors have been covered by the previous transmission. The dominant pruning algorithm uses 2-hop neighborhood information. The *forward node list* is selected in such a way that they cover all the nodes within two hops. A similar forward node selection algorithm, *multipoint relaying*, is proposed in [14].

Ni et al. [9] discuss the broadcast storm problem. They also analyze broadcast redundancy, contention, and collision in blind flooding. Algorithms for reducing broadcast redundancy are proposed, such as probabilistic scheme, counter-based scheme, distance-based scheme, etc. All of these algorithms require that each forward node estimates network redundancy and accumulates information about the network to assist its decision. Since all of these approaches are probabilistic in nature, they cannot guarantee all the nodes in the network receive the broadcast packet.

Peng and Lu propose a scalable broadcast algorithm in [12]. Similar to the self-pruning algorithm, a node does not rebroadcast the broadcast packet if all of its neighbors have received the packet from previous transmissions (not *the* previous transmission as in self-pruning). A random delay is associated with each node, measuring the time between receiving the packet for the first time and making a rebroadcast decision.

In [16], Stojmenovic et al. study a connected-dominant-set-based broadcast algorithm that uses only internal nodes to forward the broadcast packet. Internal nodes are dominating nodes derived by Wu and Li's *marking process* [17]. That is, nodes that are not internal nodes only receive the broadcast packet without forwarding it. Therefore, the number of redundant transmissions is reduced.

Calinescu et al. [2] propose a location-aware pruning method that extends the work of Lim and Kim. It is shown that the resultant dominating set has a constant approximation ratio of six. In our paper, we assume that each host has no location information of other hosts and we will compare with only those protocols that do not depend on location information.

Note that extensive work has been done in the theoretical community on finding a good approximation of minimum connected dominating set (MCDS) in terms of small *approximation ratio*. In fact, a protocol with a constant approximation ratio of eight has recently been proposed without using location information [1]. However, this approach is based on a global infrastructure (spanning tree) to select dominating nodes. It is overkill to first construct a spanning tree, select dominating nodes (forward nodes) from the tree, and then perform a broadcast. Our approach is based on constructing a connected dominating set "on-the-fly" and it is suitable for dynamic networks with mobile hosts.

3 Preliminaries

We use a simple graph, $G = (V, E)$, to represent an ad hoc wireless network, where V represents a set of wireless mobile hosts (nodes) and E represents a set of edges. An edge (u, v) indicates that both hosts u and v are within their transmitter ranges and, hence, the connections of hosts are based on geographic distances of hosts. Such a graph is also called a *unit disk graph* [3]. The circle around a host u corresponds to the transmitter range of host u. All the hosts in the circle are considered the neighbors of host u. A host can obtain its neighborhood information by periodically sending an update message. Another efficient way uses the piggyback technique; that is, when a host needs to send a packet, it attaches its neighborhood information along with the packet. We use $N(u)$ to represent the neighbor set of u (including u). $N(N(u))$ represents the neighbor set of $N(u)$ (i.e., the set of nodes that are within two hops from u). Clearly, $\{u\} \subseteq N(u) \subseteq N(N(u))$ and, if $u \in N(v)$, then $N(u) \subseteq N(N(v))$. Note that 2-hop neighborhood information can be obtained by periodic "Hello" packets, each of which contains the sender's identification and the list of its neighbors. Throughout the paper, we assume that u (sender) and v (receiver) are neighbors.

3.1 The Approximation of MCDS (AMCDS) Algorithm

As mentioned earlier, finding the minimum number of forward nodes is the same as finding a minimum connected dominating set (MCDS) in a network. Since this is an NP-complete problem, we use an approximation algorithm AMCDS proposed in [4]. At the start of the algorithm, all nodes are colored white and, then, the node with the maximum node degree is selected (put in set C) and colored black, and all of its neighbors are colored gray. A recursive selection process runs until no white node exists: Choose a gray node that has the maximum number of white neighbors. Color the selected node black and its white

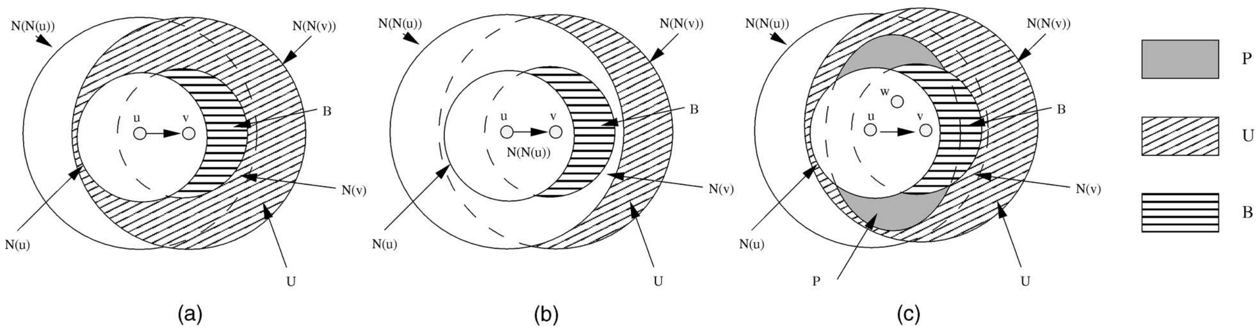


Fig. 2. Illustration for three algorithms: (a) Dominant pruning (DP), (b) total dominant pruning (TDP), and (c) partial dominant pruning (PDP).

neighbors gray. The resultant node set C is an approximation for the MCDS. The drawback of this algorithm is that it needs to know the global network topology and, therefore, it is not suitable for ad hoc wireless networks. However, we use the result of the AMCDS algorithm as the lower bound for the MCDS to compare with the results from other approximation approaches.

3.2 The Dominant Pruning (DP) Algorithm

Selection process [7]:

1. Let $F(u,v) = [\,]$ (empty list), $Z = \phi$ (empty set), and $K = \cup S_i$, where $S_i = N(v_i) \cap U(u,v)$ for $v_i \in B(u,v)$.
2. Find set S_i whose size is maximum in K. (In case of a tie, the one with the smallest identification i is selected.)
3. $F(u,v) = F(u,v)||v_k$, $Z = Z \cup S_i$, $K = K - S_i$, and $S_j = S_j - S_i$ for all $S_j \in K$.
4. If $Z = U(u,v)$, exit; otherwise, goto step 2.

As indicated in [7], the DP algorithm shows a better performance compared with other flooding algorithms such as blind flooding and self-pruning. In the DP algorithm, when node v receives a packet from node u, it selects a minimum number of forward nodes that can cover all the nodes in $N(N(v))$. Among nodes in $N(N(v))$, u is the source node, nodes in $N(u)$ have already received the packet, and nodes in $N(v)$ will receive the packet after v rebroadcasts the packet. Note that $N(u)$ can be directly derived from $N(N(v))$ once node v knows the sender identification of u. Therefore, v just needs to determine its forward node list $F(u,v)$ from $B(u,v) = N(v) - N(u)$ to cover nodes in $U(u,v) = N(N(v)) - N(u) - N(v)$. ($U(u,v)$ is the area with oblique lines in Fig. 2.) Specifically, the *greedy set cover* algorithm [8] is used for the selection of forward nodes. $F(u,v) = [f_1, f_2, ..., f_m]$, with $f_i \in B(u,v)$ satisfying $\cup_{f_i \in F}(N(f_i) \cap U(u,v)) = U(u,v)$, is derived by repeatedly selecting f_i that has the maximum number of uncovered neighbors in $U(u,v)$. The above process is called the *selection process*.[1] Z is a subset of $U(u,v)$ covered so far. S_i is the neighbor set of v_i in $U(u,v)$. K is the set of S_i. In subsequent discussion, $U(u,v)$, $B(u,v)$, and $F(u,v)$ are denoted as U, B, and F, respectively.

Dominant Pruning (DP) algorithm [7]:

1. Node v uses $N(N(v))$, $N(u)$, and $N(v)$ to obtain
$$U(u,v) = N(N(v)) - N(u) - N(v)$$
and
$$B(u,v) = N(v) - N(u).$$
2. Node v then calls the selection process to determine $F(u,v)$.

1. The DP algorithm may not terminate using the selection process, that is, $N(B(u,v))$ cannot cover $U(u,v)$. For the DP algorithm, Step 4 of the selection process should be changed to: If no new node is added to Z, exit; otherwise, goto step 2.

4 Enhanced Dominant Pruning Algorithms

In this section, we first propose two enhanced dominant pruning algorithms: the total dominant pruning (TDP) algorithm and the partial dominant pruning (PDP) algorithm. Both algorithms are then illustrated through an example.

4.1 The Total Dominant Pruning (TDP) Algorithm

If node v can receive a packet piggybacked with $N(N(u))$ from node u, the 2-hop neighbor set that needs to be covered by v's forward node list F is reduced to $U = N(N(v)) - N(N(u))$. The total dominant pruning (TDP) algorithm uses the above method to reduce the size of U and, hence, to reduce the size of F.

Total Dominant Pruning (TDP) algorithm:

1. Node v uses $N(N(v))$, $N(N(u))$, $N(u)$, and $N(v)$ to obtain
$$U = N(N(v)) - N(N(u))$$
and
$$B = N(v) - N(u).$$
2. Node v then calls the selection process to determine F.

The correctness of excluding $N(N(u))$ from $N(N(v))$ in U is shown in the following theorem.

Theorem 1. *If a node $w \in N(N(v))$ is also in $N(N(u))$, then w can be excluded from U.*

Proof. Note the fact that nodes in U are those that need to be covered by v's forward nodes. Suppose $w \in N(N(v))$, if w is in $N(N(u))$, then 1) w is in $N(u)$ (including w is v itself), 2) w is not in $N(u)$ and u uses v as a forward node to cover w, or 3) w is covered not by v, but by another neighbor of u. Obviously, for cases 1) and 3), w can be excluded from U. For case 2), w can be directly covered by v. Therefore, w can also be excluded from U. □

The fact that forward nodes can be selected from B to cover U in the TDP algorithm is shown in the following theorem.

Theorem 2. *Let* $U = N(N(v)) - N(N(u))$ *and*

$$B = N(v) - N(u),$$

then, $U \subseteq N(B)$.

Proof. Using the fact that $N(X) - N(Y) \subseteq N(X - Y)$, where X and Y are two sets. For any $w \in N(N(v)) - N(N(u))$, we have $w \in N(N(v) - N(u))$. Therefore, $N(B) = N(N(v) - N(u))$ can cover $U = N(N(v)) - N(N(u))$. □

The extra cost of the TDP algorithm is that 2-hop neighborhood information of each sender is piggybacked in the broadcast packet. Therefore, it consumes more bandwidth.

4.2 The Partial Dominant Pruning (PDP) Algorithm

In the partial dominant pruning (PDP) algorithm, like the DP algorithm, no neighborhood information of the sender is piggybacked with the broadcast packet. Therefore, the deduction of $N(N(u))$ from $N(N(v))$ cannot be done at node v. However, besides excluding $N(u)$ and $N(v)$ from $N(N(v))$, as addressed in the DP algorithm, more nodes can be excluded from $N(N(v))$. These nodes are the neighbors of each node in $N(u) \cap N(v)$. Such a node set is donated as $P(u,v)$ (or simply P) = $N(N(u) \cap N(v))$. Therefore, the 2-hop neighbor set U in the PDP algorithm is $U = N(N(v)) - N(u) - N(v) - P$. Note that, since $P = N(N(u) \cap N(v)) \subseteq N(N(u))$, Theorem 1 guarantees that P can be excluded from $N(N(v))$. The fact that forward nodes can be selected from B to cover U in the PDP algorithm is shown in the following theorem.

Theorem 3. *Let* $P = N(N(u) \cap N(v)), U = N(N(v)) - N(u) - N(v) - P$ *and* $B = N(v) - N(v)$, *then* $U \subseteq N(B)$.

Proof. Since $N(N(v)) - N(u) - N(v) - N(N(u) \cap N(v)) \subseteq N(N(v)) - N(N(u) \cap N(v))$ and the fact that $N(X) - N(X \cap Y) \subseteq N(X - (X \cap Y)) = N(X - Y), N(B) = N(N(v) - N(u))$ can cover $N(N(v)) - N(N(u) \cap N(v))$ and, hence, can cover

$$U = N(N(v)) - N(u) - N(v) - N(N(u) \cap N(v)).$$

□

Partial Dominant Pruning (PDP) algorithm:

1. Node v uses $N(N(v))$, $N(u)$, and $N(v)$ to obtain

$$P = N(N(u) \cap N(v)),$$
$$U = N(N(v)) - N(u) - N(v) - P,$$

and

$$B = N(v) - N(u).$$

2. Node v then calls the selection process to determine F.

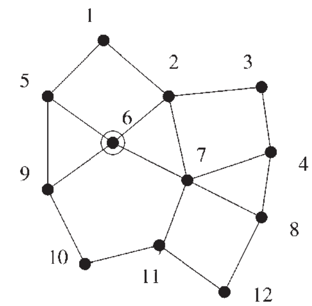


Fig. 3. A sample network of 12 nodes with source node 6.

While the PDP algorithm does not increase the size of the broadcast packet, compared with the DP algorithm, it eliminates more redundant transmissions. The only additional computational cost for the PDP algorithm is that each forward node v needs to calculate set P.

Like the DP, both the TDP and PDP do not have a constant approximation ratio, although both work well in the average case, as confirmed by the simulation results shown in Section 6. However, both the TDP and PDP can be extended to a clustered network where some clusterheads are selected as forward nodes. It is shown in [18] that a constant approximation ratio can be achieved by using the pruning technique in the clustered network.

Note that, although excessive broadcast redundancy will cause the broadcast storm problem, some broadcast redundancy in the ad hoc wireless network could be useful to ensure a high *broadcast delivery rate*, especially when a host cannot update its neighborhood information (1-hop and 2-hop neighbor sets) in a timely manner. The broadcast delivery rate is defined as the number of hosts that receive the packet over the total number of hosts in the network. Consider a case when u forwards a broadcast packet to v. Suppose w that was in the coverage area (within two hops) of v moves out and enters the coverage area of u before u and v update their neighborhood information. If w is selected as a forward node by v, then nodes covered by w in the coverage area of v may miss the packet unless they are covered by other nodes (if the situation exists, depending on the network topology and broadcast redundancy). Even if w is not selected as a forward node by v, w itself may miss the packet when 1) it enters the coverage area of u after the broadcast within the coverage area of u has completed or 2) it enters the coverage area of u before the broadcast within the coverage area of u completes, but no forward node selected by u can cover w. In the absence of contention and collision, the broadcast delivery rate depends on how frequently the neighborhood information can be updated (relative to the moving speed of mobile hosts). Reliable broadcast that guarantees delivery is a totally different and complex issue and it needs a special treatment. The traditional hop-by-hop or end-to-end acknowledgment (both positive and negative) can be applied, but it is expensive to enforce. Another option is for each host to keep the received broadcast packet for a certain period, it will unicast the packet to any new host that enters its coverage area. In Section 5, a special environment is defined such that the broadcast process can guarantee to deliver the broadcast packet to each host.

TABLE 1
Neighbors Within Two Hops (Fig. 3)

v	$N(v)$	$N(N(v))$
1	1,2,5	1,2,3,5,6,7,9
2	1,2,3,6,7	1,2,3,4,5,6,7,8,9,11
3	2,3,4	1,2,3,4,6,7,8
4	3,4,7,8	2,3,4,6,7,8,11,12
5	1,5,6,9	1,2,5,6,7,9,10
6	2,5,6,7,9	1,2,3,4,5,6,7,8,9,10,11
7	2,4,6,7,8,11	1,2,3,4,5,6,7,8,9,10,11,12
8	4,7,8,12	2,3,4,6,7,8,11,12
9	5,6,9,10	1,2,5,6,7,9,10,11
10	9,10,11	5,6,7,9,10,11,12
11	7,10,11,12	2,4,6,7,8,9,10,11,12
12	8,11,12	4,7,8,10,11,12

4.3 Example

Fig. 3 shows a sample network of 12 nodes with source node 6. Neighborhood information of each node is shown in Table 1. We illustrate different forward node lists for these three algorithms.

For the DP algorithm, nodes in $N(6)$ will receive the packet directly. Since

$$U(\phi, 6) = N(N(6)) - N(6) = \{1, 3, 4, 8, 10, 11\},$$

the forward node list for node 6 is $F(\phi, 6) = [7, 2, 9]$. (The selection order is 7, 2, and 9.) From

$$U(6, 7) = N(N(7)) - N(6) - N(7) = \{1, 3, 10, 12\},$$

we have $F(6, 7) = [11, 4]$. Similarly, from

$$U(6, 2) = N(N(2)) - N(6) - N(2) = \{4, 8, 11\},$$

we have $F(6, 2) = [3]$; from

$$U(6, 9) = N(N(9)) - N(6) - N(9) = \{1, 11\},$$

we have $F(6, 9) = [10]$. Therefore, the total number of forward nodes (including the source node) is $1 + 3 + 4 = 8$.

For the TDP algorithm, node 6 has the same forward node list $F(\phi, 6) = [7, 2, 9]$. From

$$U(6, 7) = N(N(7)) - N(N(6)) = \{12\},$$

we have the forward node list for node 7: $F(6, 7) = [8]$. Similarly, from $U(6, 2) = N(N(2)) - N(N(6)) = \phi$, we have $F(6, 2) = [\,]$; from $U(6, 9) = N(N(9)) - N(N(6)) = \phi$, we have $F(6, 9) = [\,]$. Therefore, the total number of forward nodes is $1 + 3 + 1 = 5$.

For the PDP algorithm, node 6 again has the same forward node list $F(\phi, 6) = [7, 2, 9]$. From $P(6, 7) = \{1, 3, 6, 7\}$, we have

$$U(6, 7) = N(N(7)) - N(6) - N(7) - P(6, 7) = \{10, 12\}.$$

The forward node list for node 7 is $F(6, 7) = [11]$. Similarly, from $P(6, 2) = \{2, 4, 6, 8, 11\}$, we have

$$U(6, 2) = N(N(2)) - N(6) - N(2) - P(2, 6) = \phi$$

and, then, $F(6, 2) = [\,]$; from $P(6, 9) = \{1, 6, 9\}$, we have

$$U(6, 9) = N(N(9)) - N(6) - N(9) - P(9, 6) = \{11\}$$

and, then, $F(6, 9) = [10]$. Therefore, the total number of forward nodes is $1 + 3 + 2 = 6$.

The details of P, U, B, and F for different broadcast algorithms are shown in Table 2, Table 3, and Table 4. From this example, we can see the performance improvement of the PDP and TDP compared with the DP in terms of generating a small number of forward nodes. As the lower bound by using the AMCDS algorithm, the minimum connected dominating set is $\{2, 6, 7, 11\}$, so the number of forward nodes is 4.

Fig. 4 shows an ad hoc wireless network in a broadcast area of 100×100. There are 80 hosts, each of which has a transmitter range of 20. The source node, forward nodes, and nonforward nodes are represented by different types of cycles. Total numbers of forward nodes are 51 for the DP, 46 for the PDP, and 44 for the TDP, respectively.

TABLE 2
The DP Algorithm

u	v	U	B	F
ϕ	6	1,3,4,8,10,11	2,5,7,9	7,2,9
6	7	1,3,10,12	4,8,11	11,4
6	2	4,8,11	1,3	3
6	9	1,11	10	10
7	11	9	10,12	10
7	4	12	3	[ ]
2	3	8	4	4
9	10	7,12	11	11

TABLE 3
The TDP Algorithm

u	v	U	B	F
ϕ	6	1,3,4,8,10,11	2,5,7,9	7,2,9
6	7	12	4,8,11	8
6	2	ϕ	1,3	[ ]
6	9	ϕ	10	[ ]
7	8	ϕ	12	[ ]

5 Termination Criteria

When a source node broadcasts a packet, each intermediate node will decide whether to rebroadcast the packet or to drop it independently, based on a given termination criterion. In

TABLE 4
The PDP Algorithm

u	v	P	U	B	F
ϕ	6	ϕ	1,3,4,8,10,11	2,5,7,9	7,2,9
6	7	1,3,6,7	10,12	4,8,11	11
6	2	2,4,6,8,11	ϕ	1,3	[ ]
6	9	1,6,9	11	10	10
7	11	ϕ	9	10,12	10
9	10	ϕ	7,12	11	11

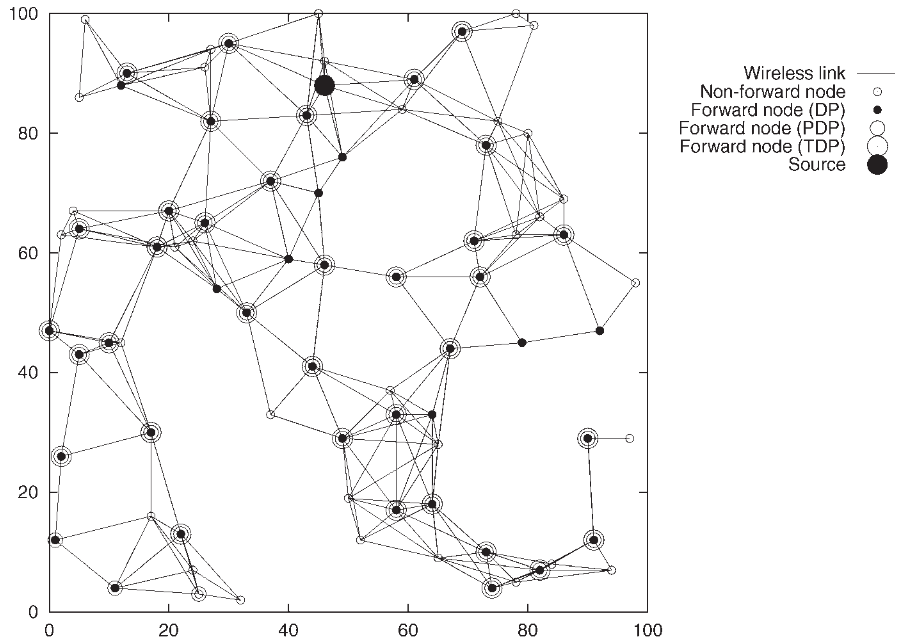


Fig. 4. Distribution of forward nodes using DP, TDP, and PDP algorithms.

other words, the broadcast process at each node will terminate when a given termination criterion is satisfied. To determine a termination criterion that guarantees delivery, we assume the following "static" environment: Mobile hosts are still allowed to roam freely in the working space. However, the broadcast process (including the forward node selection and the broadcast process itself) is done quickly so that $N(v)$ and $N(N(v))$ remain the same during the process for each host v. In addition, each host v has updated and consistent $N(v)$ and $N(N(v))$ when the broadcast process starts.

Here, two criteria are used to determine the termination of a broadcast process. The first one assigns a *marked/unmarked* status to each node. A node v is called *marked* if v has received a packet; otherwise, v is called *unmarked*. We assume that, v knows the current marked/unmarked status of the nodes in $N(v)$ at the time v decides its forward node list. When all nodes in $N(v)$ are marked, v will stop rebroadcasting and discard the packet. Since each node needs to keep track of changing status information of neighbors, it is a relatively expensive approach. The following theorem shows that such a criterion is sufficient for v to guarantee that all nodes in $N(N(v))$ can receive the broadcast packet.

Theorem 4. *Using the marked/unmarked termination criterion, all nodes in the network will be marked upon termination.*

Proof. The node set can be covered by a set of forward nodes (including the source) and their 2-hop neighbor sets. We prove the following: 1) If a forward node u is marked, all nodes in $N(N(u))$ will eventually be marked. 2) All forward nodes will be marked once the source initiates the broadcast process.

Proof for 1): Referring to Fig. 5, we arbitrarily select a forward node u in the network (the forward node set differs from algorithm to algorithm). If u does forward the broadcast packet, the claim is clearly true; otherwise, u stops because all of its neighbors have been marked. In the latter case, we show that all 2-hop neighbors of u (i.e., nodes in $N(N(u)) - N(u)$) are marked upon termination. Arbitrarily select w from 2-hop neighbors of u and select v such that $w \in N(v)$ and $v \in N(u)$. Suppose v is first marked by u' (i.e., $v \in N(u')$ and, hence, $w \in N(N(u'))$), we consider the following two cases:

1. If v is a forward node for u' in $N(N(u'))$; clearly, w will be marked by v (if no other node does it first).
2. If v is not a forward node for u' in $N(N(u'))$, then assume that v' is a forward node for u' in $N(N(u'))$ that covers w (i.e., $w \in N(v')$). The fact that u' marked v for the first time means that u' did send out the broadcast packet to all its neighbors, including v'. v' will mark w if w is not marked by any other neighbors of w.

Proof for 2): Note that the subgraph induced from the forward node set (which includes the source) is a connected graph. Starting from the source which is marked initially, iteratively applying the above result

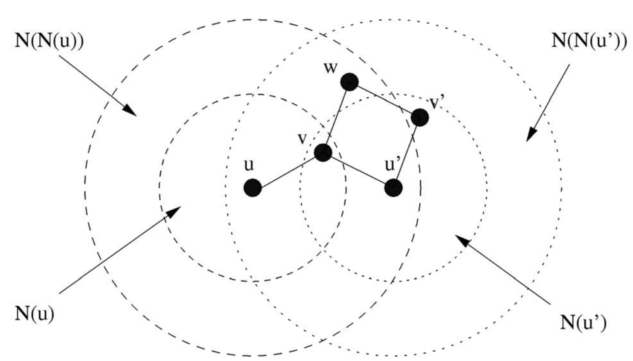


Fig. 5. Marked/unmarked termination criterion.

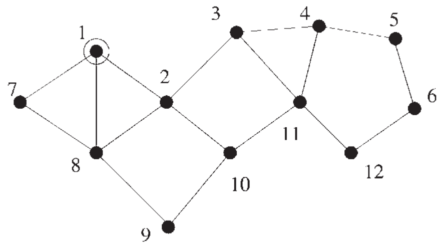


Fig. 6. An illustrative example for different termination criteria.

TABLE 5
An Illustrative Example for Different Termination Criteria

algorithm	marked/un-marked	relayed/un-relayed
DP	1,2,3,4,5,8,9,10,11,12	1,2,3,4,5,6,8,9,10,11,12
PDP	1,2,3,4,5,8,11,12	1,2,3,4,5,6,8,10,11,12
TDP	1,2,3,4,5,8,11,12	1,2,3,4,5,8,10,11,12

1), we will eventually mark all the nodes in the forward node set. □

The second approach assigns a *relayed/unrelayed* status to each node. A node v is called *relayed* when v has sent a packet; otherwise, v is called *unrelayed*. Forward node v will stop rebroadcasting a packet only when v has sent that packet. The correctness of this approach is apparent. In general, more nodes will be selected as the forward nodes in this approach compared with the first approach. Since each termination is decided locally, this approach corresponds to a reasonable termination criterion in a real system. Note that a relayed node must be a marked node, but not vice versa.

Referring to Fig. 6, suppose the source is node 1, forward node sets with two termination criteria are shown in Table 5. Generally, the number of forward nodes of the marked/unmarked termination criterion is less than that of the relayed/unrelayed termination criterion.

6 Performance Simulations

We simulate the performance of the DP, PDP, and TDP algorithms in terms of the average number of forward nodes generated. The simulation is conducted under the static environment defined earlier. The simulator randomly generates a connected unit disk graph within a broadcast area of $m \times m$ (with $m = 100$). Graphs are generated in two ways: a fixed transmitter range (r) and a fixed average node degree (d). The number of hosts ranges from 20 to 100. For each given number of hosts, 400 random graphs are generated. An ideal MAC layer is assumed so that no contention or collision will occur.

Fig. 7, Fig. 8, Fig. 9, and Fig. 10 show the average numbers of forward nodes and Fig. 11, Fig. 12, Fig. 13, and Fig. 14 show the average numbers of packets a node receives during the broadcast process under different algorithms and termination criteria. We check the effect of node transmitter range and average node degree on the performance of these algorithms. These two parameters are indeed related to each other: The average node degree is the expected number of nodes (out of n) that are within a node's transmitter range. Specifically, the average node degree can be approximated as $d = (\frac{\pi r^2}{m^2})n$, where r is the transmitter range and m is the length of each side of the confined working space. This approximation is fairly accurate, especially when $r \ll m$. Basically, we measure the same feature from two different viewpoints and obtain the most sensitive parameter under various simulations.

Fig. 7 and Fig. 8 show the simulation results of the average number of forward nodes for fixed transmitter ranges (from 25 to 70), under both marked/unmarked and relayed/unrelayed termination criteria. Fig. 9 and

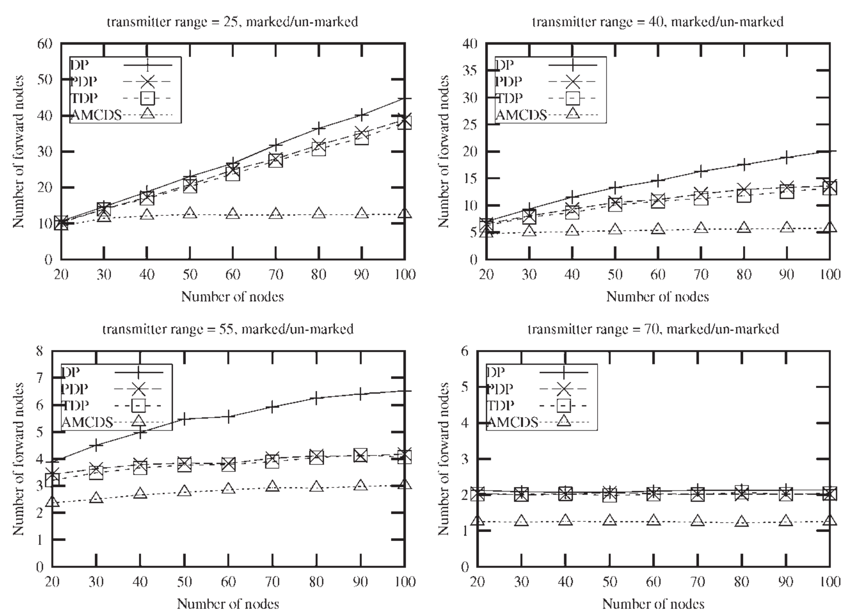


Fig. 7. The average number of forward nodes with the marked/unmarked termination criterion with fixed transmitter ranges from 25 to 70.

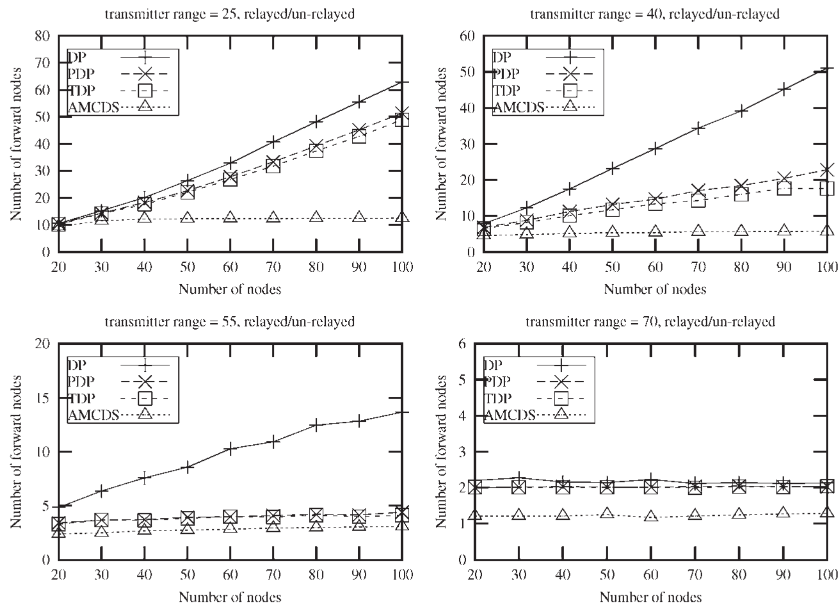


Fig. 8. The average number of forward nodes with the relayed/unrelayed termination criterion with fixed transmitter ranges from 25 to 70.

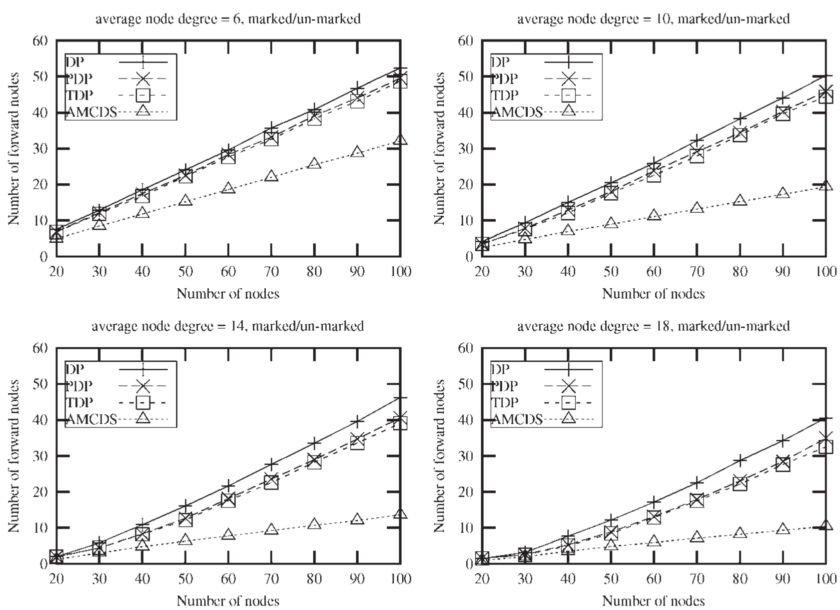


Fig. 9. The average number of forward nodes with the marked/unmarked termination criterion with fixed average node degrees from 6 to 18.

Fig. 10 show simulation results of the average number of the forward nodes for fixed node degrees (from 6 to 18), under both marked/unmarked and relayed/unrelayed termination criteria. From these simulation results, we can conclude that both the TDP and PDP have better performance than the DP in both fixed-transmitter-range networks and fixed-node-degree networks. When the transmitter range is 25, the percentages of the reduced forward nodes based on the PDP and TDP compared with that of the DP are 15 percent under the marked/unmarked termination criterion and are almost 20 percent when the relayed/unrelayed termination criterion is applied. The result of the TDP is a range from 2 percent to 5 percent lower than that of the PDP. We can see that, when the transmitter range increases, the number of forward nodes drops. In addition, the number of forward nodes is directly affected by the node degree since it is linearly proportional to the node degree, as shown in Fig. 10. The results for the TDP and PDP are very close in all cases. Therefore, the PDP is more cost effective since no neighborhood information of the sender is piggybacked in the PDP during the transmission.

We use the result from the AMCDS algorithm as the lower bound to evaluate the effectiveness of each algorithm. Clearly, the result using local 2-hop neighborhood information still cannot match the one using the global network information. However, results from the PDP and TDP are close to the lower bound when the network has either a large transmitter range or a large node degree. The simulation also shows that the difference between two

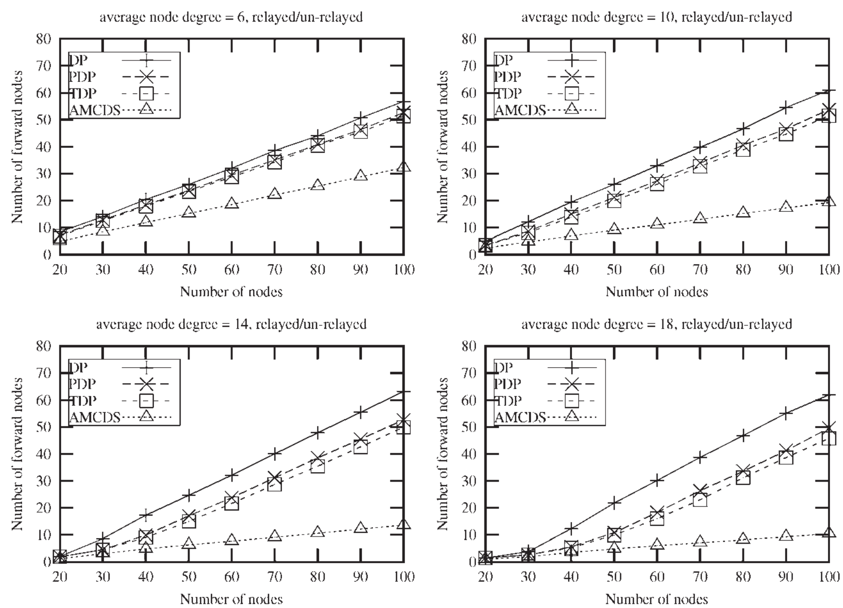


Fig. 10. The average number of forward nodes with the relayed/unrelayed termination criterion with fixed average node degrees from 6 to 18.

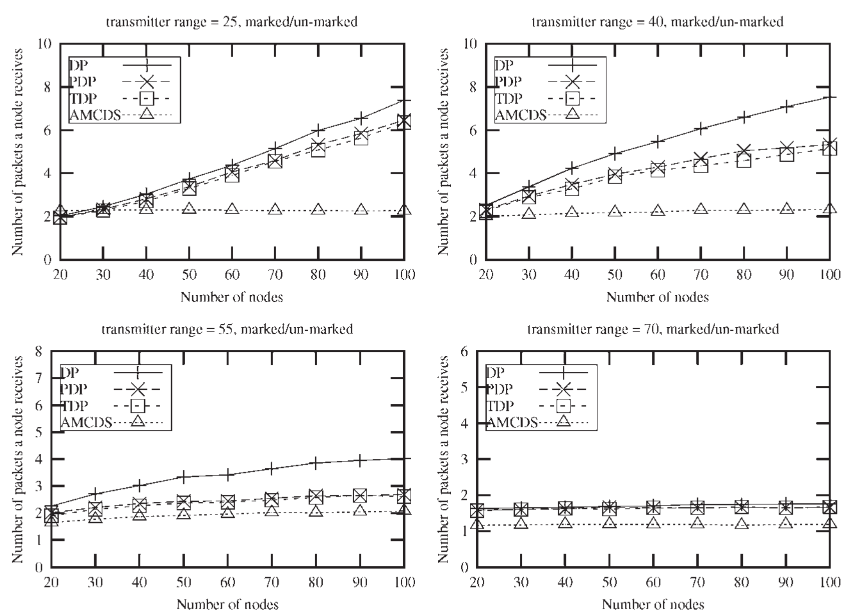


Fig. 11. The average number of packets a node receives with the marked/unmarked termination criterion with fixed transmitter ranges from 25 to 70.

termination criteria exists and becomes significant when the number of nodes increases. The performance using marked/unmarked status is better than the one using relayed/unrelayed status because, in the latter, a node v may not be able to detect on time that all the nodes in $N(v)$ have already received the packet.

Fig. 11 and Fig. 12 show the simulation results of the average number of broadcast packets that a node receives during the broadcast process for fixed transmitter ranges (from 25 to 70), under both marked/unmarked and relayed/unrelayed termination criteria. Fig. 13 and Fig. 14 show the simulation results of the average number of broadcast packets that a node receives during the broadcast process for fixed node degrees (from 6 to 18), under both marked/unmarked and relayed/unrelayed termination criteria. These figures show the degree of redundancy, which is vital to ensure a high broadcast delivery rate when neighborhood information cannot be updated in a timely manner. From these simulations (from Fig. 11 to Fig. 14), we can see that differences among these algorithms exist in terms of broadcast redundancy (i.e., the average number of broadcast packets a node receives). This is not surprising, because the degree of broadcast redundancy directly relates to the number of the forward nodes. The more the number of forward nodes in a broadcast process, the higher the broadcast redundancy.

A separate simulation for the "dynamic" environment is conducted. In this simulation, as in the static environment, the broadcast process is still assumed to complete quickly so that both $N(v)$ and $N(N(v))$ remain the same during the process for each host v. However, v cannot update its $N(v)$ and $N(N(v))$ in a timely and consistent manner because

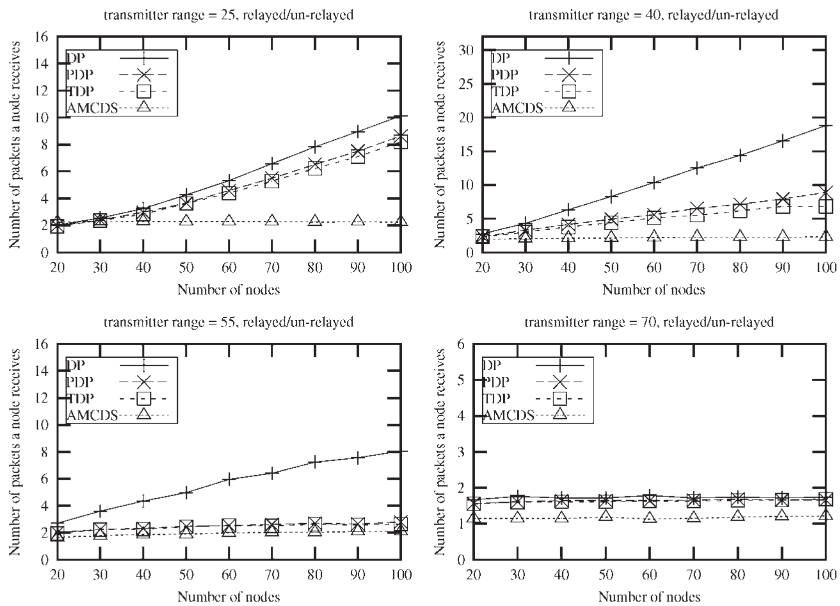


Fig. 12. The average number of packets a node receives with the relayed/unrelayed termination criterion with fixed transmitter ranges from 25 to 70.

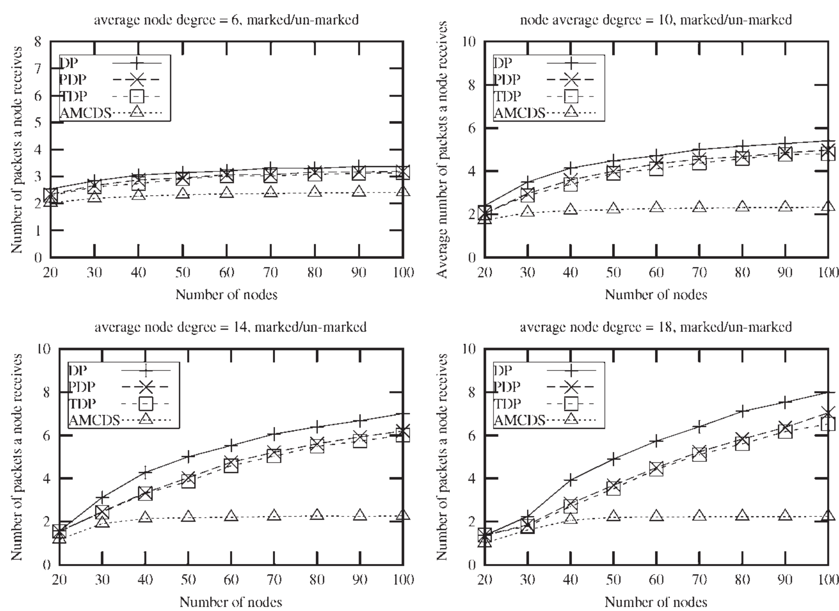


Fig. 13. The average number of packets a node receives with the marked/unmarked termination criterion with fixed average node degrees from 6 to 18.

mobile hosts are moving at a fast speed. Specifically, it is assumed that neighborhood information is updated at each time unit. The broadcast may occur at any time with a uniform distribution and completes quickly (so we can assume no host movement during the broadcast process). Each node follows the random walk model by selecting a destination and moves toward it with a randomly selected speed. A new destination is selected (together with a new randomly selected speed) when the current one is reached. The maximum speed ranges from 5 to 75 per time unit. The actual speed is uniformly selected between zero and the maximum speed. The simulation runs 1,000 times and generates an average value for each case. Note that the speed of a host movement, slow, moderate, or fast, is relative to the transmitter range of the node. Specifically, when the distance of a host movement per time unit is significantly less than the transmitter range, the host is said to be in a slow movement; when the distance of a host movement per time unit is significantly more than the transmitter range, the host is said to be in a fast movement. Between the slow and fast movement is the moderate movement. For example, for the case that the transmitter range of the node is 25, 25 is considered a moderate speed because, between two neighborhood information updates, a host can move out of the transmitter range; while 75 is considered a high speed because a host can move three times the transmitter range. It is commonly agreed that if all the hosts in the network move in a fast speed, there is no good solution other than blind flooding.

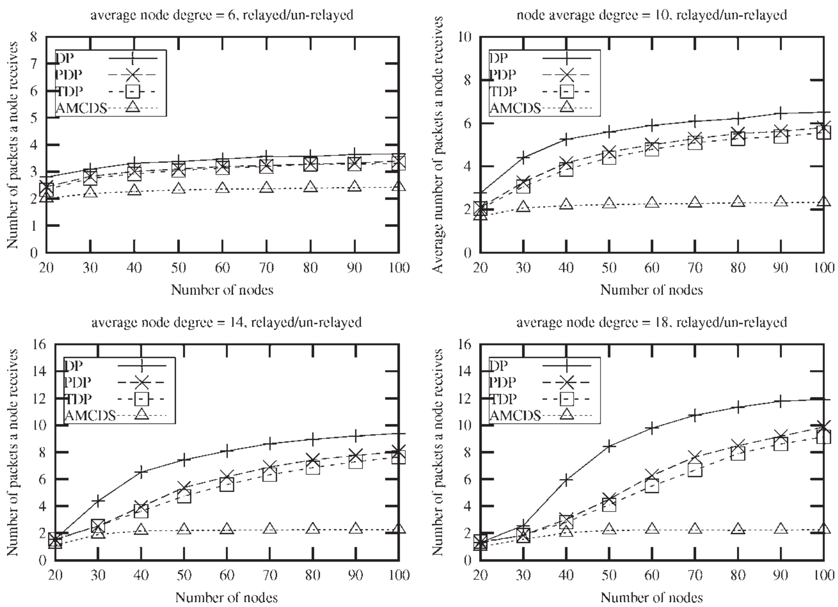


Fig. 14. The average number of packets a node receives with the relayed/unrelayed termination criterion with fixed average node degrees from 6 to 18.

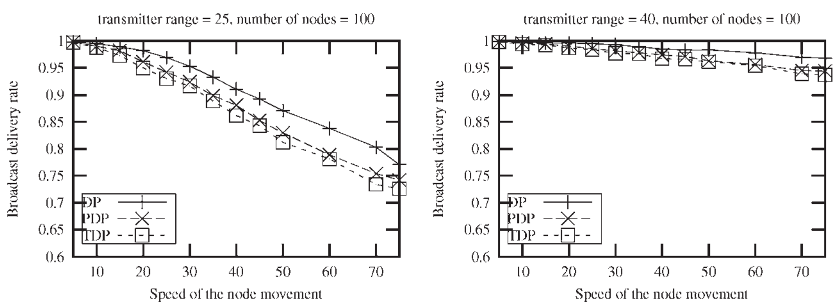


Fig. 15. The broadcast delivery rate when the transmitter ranges are 25 to 40.

Fig. 15 shows the broadcast delivery rate when the transmitter ranges are 25 and 40, respectively. In Fig. 15, the broadcast delivery rate decreases as the speed of each host increases. The rate also depends on the degree of broadcast redundancy, the DP has the highest, followed by the PDP, and the TDP has the lowest. The difference among three algorithms, in terms of the broadcast delivery rate, is less than 5 percent. All three algorithms ensure relatively high broadcast delivery rates (over 90 percent) in the given ranges of speed from slow to moderate.

7 Conclusions

In this paper, we have studied the broadcast process in ad hoc wireless networks with an objective of minimizing the number of forward nodes. We have pointed out the deficiencies of the dominant pruning (DP) algorithm and proposed two new algorithms: the total dominant pruning (TDP) and the partial dominant pruning (PDP). Given the sender u and receiver v, the TDP uses $N(N(u))$ and $N(N(v))$ to obtain a smaller 2-hop neighbor set $U_{TDP} = N(N(v)) - N(N(u))$ that needs to be covered by v's forward nodes. The PDP uses $N(u)$ and $N(v)$ to eliminate more nodes from $N(N(v))$ compared with the DP. Nodes in $P = N(N(u) \cap N(v))$ can be excluded from $N(N(v))$. Specifically, $U_{DP} = N(N(v)) - N(u) - N(v)$ and $U_{PDP} = U_{DP} - P$. Clearly, $U_{TDP} \subseteq U_{PDP} \subseteq U_{DP}$. Simulation results have shown that both proposed algorithms have better performance than the original DP algorithm and the difference between the TDP and PDP is insignificant. The relationship between broadcast redundancy and broadcast delivery rate has also been studied and simulated under the dynamic environment. All three algorithms ensure a high broadcast delivery rate when the host movement ranges from slow to moderate. In addition, the difference between the DP (the best) and the TDP (the worst) is less than 5 percent. We have also discussed two termination criteria and shown that the practical termination criterion can also obtain satisfactory results. One direction of future work is to extend the proposed scheme from a coverage area of 2-hop to k-hop.

Acknowledgments

This work was supported in part by US National Science Foundation grants CCR 9900646 and ANI 0073736. We would like to give our thanks to Ellie Shuff for proofreading our paper and giving helpful comments.

REFERENCES

[1] K.M. Alzoubi, P.J. Wan, and O. Frieder, "New Distributed Algorithm for Connected Dominating Set in Wireless Ad Hoc Networks," *Proc. Hawaii Int'l Conf. System Sciences '35,* 2002.

[2] G. Calinescu, I. Mandoiu, P.J. Wan, and A. Zelikovsky, "Selecting Forwarding Neighbors in Wireless Ad Hoc Networks," *Proc. ACM Int'l Workshop Discrete Algorithms and Methods for Mobile Computing (DIALM '01),* pp. 34-43, Dec. 2001.

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

[4] B. Das, R. Sivakumar, and V. Bharghavan, "Routing in Ad-Hoc Networks Using a Virtual Backbone," *Proc. Int'l Conf. Computer Comm. and Networks '97,* pp. 1-20, Sept. 1997.

[5] C. Ho, K. Obraczka, G. Tsudik, and K. Viswanath, "Flooding for Reliable Multicast in Multihop Ad Hoc Networks," *Proc. ACM Int'l Workshop Discrete Algorithms and Methods for Mobile Computing '99,* pp. 64-71, Aug. 1999.

[6] D.B. Johnson and D.A. Maltz, *Mobile Computing,* chapter "Dynamic Source Routing in Ad-Hoc Wireless Networks," pp. 153-181, Kluwer Academic, 1996.

[7] H. Lim and C. Kim, "Flooding in Wireless Ad Hoc Networks," *Computer Comm. J.,* vol. 24, no. 3-4, pp. 353-363, 2001.

[8] L. Lovasz, "On the Ratio of Optimal Integral and Fractional Covers," *Discrete Math.,* vol. 13, pp. 383-390, 1975.

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

[10] V.D. Park and M. S. Corson, "Temporally-Ordered Routing Algorithm (TORA) Version 1: Functional Specification," *Internet Draft,* 1997.

[11] M.R. Pearlman and Z.J. Haas, "Determining the Optimal Configuration of the Zone Routing Protocol," *IEEE J. Selected Areas in Comm.,* vol. 17, no. 8, pp. 1395-1414, Feb. 1999.

[12] W. Peng and X.C. Lu, "On the Reduction of Broadcast Redundancy in Mobile Ad Hoc Networks," *Proc. First Ann. Workshop Mobile and Ad Hoc Networking and Computing, MOBIHOC,* pp. 129-130, Aug. 2000.

[13] C. Perkins and E.M. Royer, "Ad-Hoc On-Demand Distance Vector Routing," *Proc. Second IEEE Workshop Mobile Computing Systems and Applications (WMCSA),* pp. 90-100, Feb. 1999.

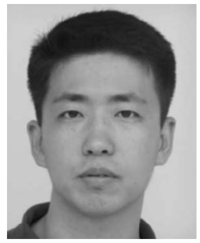
[14] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint Relaying for Flooding Broadcast Message in Mobile Wireless Networks," *Proc. Hawaii Int'l Conf. System Sciences '35,* Jan. 2002.

[15] E.M. Royer and C.K. Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks," *IEEE Personal Comm.,* vol. 6, no. 2, pp. 46-55, 1999.

[16] I. Stojmenovic, S. Seddigh, and J. Zunic, "Dominating Sets and Neighbor Elimination Based Broadcasting Algorithms in Wireless Networks," *IEEE Trans. Parallel and Distributed Systems,* vol. 13, no. 1, pp. 14-25, Jan. 2002.

[17] J. Wu and H. Li, "On Calculating Connected Dominating Sets for Efficient Routing in Ad Hoc Wireless Networks," *Proc. ACM Int'l Workshop Discrete Algorithms and Methods for Mobile Computing '99,* pp. 7-14, Aug. 1999.

[18] J. Wu and W. Lou, "Forward-Node-Set-Based Broadcast in Clustered Mobile Ad Hoc Networks." Technical Report CSE-02-15, June 2002.



Wei Lou (S'02) received the BE degree in electrical engineering from Tsinghua University, Beijing, China, in 1995 and the ME degree in telecommunications and electrical systems from Beijing University of Posts and Telecommunications, Beijing, China, in 1998. He is currently a PhD candidate in computer engineering at Florida Atlantic University, Boca Raton. Prior to attending Florida Atlantic University in 2000, he worked as a software engineer at Beijing Yinhe Computer Company. His research interests include routing protocol design, analysis, and implementation for mobile ad hoc networks. He is a student member of the IEEE.

Jie Wu is a professor at the Department of Computer Science, Florida Atlantic University. He has published more than 150 papers in various journals and conference proceedings. His research interests are in the areas of mobile computing, routing protocols, fault-tolerant computing, and interconnection networks. Dr. Wu has served on many conference organization committees. He serves on the editorial board of *IEEE Transactions on Parallel and Distributed Systems* and was a coguest editor of the *Journal of Parallel and Distributing Computing*. He is the author of the text *Distributed System Design*, (CRC press). Dr. Wu was the recipient of the 1996-1997 and 2001-2002 Researcher of the Year Award at Florida Atlantic University. He served as an IEEE Computer Society Distinguished Visitor. Dr. Wu is a member of ACM and a senior member of the IEEE.