Mobility Management and Its Applications in Efficient Broadcasting in Mobile Ad Hoc Networks

Jie Wu and Fei Dai
Department of Computer Science and Engineering
Florida Atlantic University
Boca Raton, FL 33431
Email: {jie, fdai}@cse.fau.edu.

***Abstract*— We study an efficient broadcast scheme in mobile ad hoc networks (MANETs). The objective is to determine a small set of forward nodes to ensure full coverage. We first study several methods to select a small forward node set assuming that the neighborhood information can be updated in a timely manner. Then we consider a general case, where each node updates its neighborhood information based asynchronously on a pre-defined frequency and node move even during the broadcast process. The virtual network constructed from local views of nodes may not be connected, its links may not exist in the physical network, and the global view constructed from collection of local views may not be consistent. In this paper, we first give a sufficient condition for connectivity at the physical network to ensure the connectivity at the virtual network. We then propose a solution using two transmission ranges to address the link availability issue. The neighborhood information as well as the forward node set are determined based on a short transmission range while the broadcast process is done on a long transmission range. The difference between these two ranges is based on the update frequency and the speed of node movement. Finally, we propose a mechanism called aggregated local view to ensure consistency of the global view. By these, we extend Wu and Dai's coverage condition for broadcasting in a network with mobile nodes. The simulation study is conducted to evaluate the coverage of the proposed scheme.** [1]

Keywords: Broadcasting, localized algorithms, mobile ad hoc networks (MANETs), mobility, simulation, system design.

I. INTRODUCTION

Broadcasting a packet to the entire network is a basic operation and has extensive applications in mobile ad hoc networks (MANETs). For example, broadcasting is used in the route discovery process in several routing protocols, when advising an error message to erase invalid routes from the routing table, or as an efficient mechanism for reliable multicast in a fast-moving MANET. In MANETs with the promiscuous receiving mode, the traditional blind flooding incurs significant redundancy as well as collision and causes the so-called broadcast storm problem [1]. Efficient broadcasting in a MANET focuses on selecting a small forward node set while ensuring broadcast coverage.

In a broadcast process, each node decides its forwarding status based on given neighborhood information, and the corresponding broadcast protocol is called *self-pruning*. In Figure 1, black (white) nodes are forward (non-forward) nodes. Each circle corresponds to a one-hop neighborhood. Any source node is a black node by default. Basically, forward nodes form a connected dominating set (CDS), where each node in the system is either in the set or the neighbor of a node in the set. That is, each white node is adjacent to at least one black neighbor. However, most existing broadcast schemes assume either the underlying network topology is static or semi-static during the broadcasting process such that the neighborhood information can be updated in a timely manner. The results in [2] show that existing static network broadcast schemes perform poorly in terms of delivery ratio when nodes are mobile. There are two sources that cause the failure of message delivery:

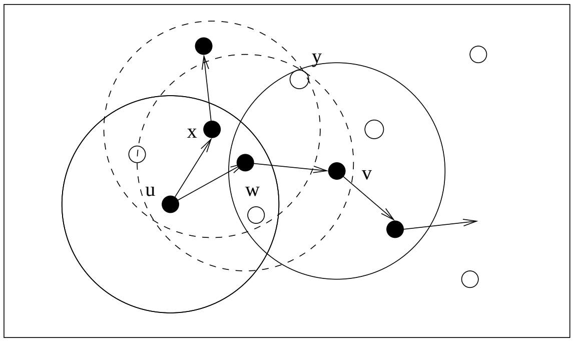


Fig. 1. Forward node set in a MANET.

Collision: The message intended for a destination collides with another message. In Figure 1, if messages from nodes w and x collide at node y, node y does not receive any message.

Mobile nodes: The neighbor in the neighbor set moves out of its transmission range (i.e., it is no longer a neighbor). In Figure 1, when node w moves out of the transmission range of u, the nodes along the branch rooted at w of the broadcast tree will miss the message[2].

---

[1] This work was supported in part by NSF grants CCR 0329741, ANI 0073736, and EIA 0130806.

[2] Nodes in the branch may still receive the message, if some adjacent nodes of the branch forward the message.

Results in [2] also show that the majority of delivery failures are caused by mobile nodes. Although many broadcast protocols have been proposed with different broadcast redundancies (and collated broadcast delivery ratios), each broadcast protocol has only its "fixed" broadcast redundancy (and broadcast delivery ratio). It is in general hard to control redundancy and delivery for a given broadcast protocol.

The major challenges in designing a localized broadcast protocol while ensuring broadcast coverage are the following: (a) The network topology changes over time, even during the broadcast process. (b) The local (1-hop) information is constructed based on "Hello" intervals. Nodes start their intervals asynchronously, making it difficult to ensure consistent local/global views among nodes. (c) The collection process for k-hop information incurs delay which may not reflect the current network topology when there are mobile nodes, even for small k in localized solutions. As a consequence, the virtual network constructed from local views of nodes may not be connected (*connectivity issue*), its links may not exist in the physical network (*link availability issue*), and the global view constructed from collection local views may not be consistent (*consistency issue*).

In this paper, we first give a sufficient condition for connectivity at the physical network to ensure the connectivity at the virtual network. We then propose a solution using two transmission ranges to address the link availability issue. The neighborhood information as well as the forward node set are determined based on a short transmission range while the broadcast process is done on a long transmission range. The difference between these two ranges is based on the update frequency and the speed of node movement. The difference is also used as a new *controllable parameter* to balance broadcast redundancy and broadcast delivery ratio. Finally, we propose a mechanism called *aggregated local view* to ensure consistency of the global view. The simulation study is conducted to evaluate the coverage of the proposed scheme. Note that the forwarding probability in probabilistic broadcasting [1] is also a controllable parameter. However, it is difficult to establish a direct connection between parameter selection and node mobility.

By providing solutions to the above three issues, we also extend Wu and Dai's coverage condition [3] for broadcasting in a network with mobile nodes. This coverage condition is a sufficient condition for a node to determine its non-forward status based on k-hop neighborhood information (for small k, say 2 or 3) only. However, the coverage condition was only suitable when the topology is static during the broadcast process and neighborhood information is consistent with the current state. Simulation results in this paper show that the proposed scheme improves the coverage significantly.

The main contributions of this paper are as follows:

1) Propose the first localized broadcast protocol that can handle mobility while ensuring broadcast coverage.
2) Systematically address the issue of inconsistent local view caused by neighborhood information delay, asynchronous "Hello" intervals, and node mobility.
3) Introduce a new controllable parameter to balance broadcast efficiency and broadcast delivery ratio.
4) Conduct a comprehensive simulation on the new approach, comparing with existing methods.

The remainder of the paper is organized as follows: Section II provides some preliminaries and related works, especially Wu and Dai's coverage condition. Section III proposes the mobility control method based on two transmission ranges, and gives some analytical study and optimization techniques. Simulation results are presented in Section IV. The paper concludes in Section V.

II. PRELIMINARIES AND RELATED WORKS

This section starts with some related work on mobility management and, in particular, neighbor set management in a mobile environment. Then an overview of broadcast protocols in MANETs based on self-pruning is given. The focus is on Wu and Dai's coverage condition and six existing protocols as its special cases.

A. Mobility management

The capacity of MANETs is constrained by the mutual interference of concurrent transmissions between nodes. The mobility of nodes adds another dimension of complexity in the mutual interference. Several studies [4], [5] focused the effect of mobility on the network capacity. Camp et al [6] gave an excellent survey on mobility models for MANETs. Three popular mobility models include (1) *random walk*, which is a simple mobility model based on random directions and speeds, (2) *random waypoint*, which includes pause time between changes in destination and speed, and (3) *random direction mobility*, which forces hosts to travel to the edge of the simulation area before changing direction and speed. In [7], a *velocity-bounded model* (for pedestrians with mobile nodes in a relatively small area), and an *acceleration-bounded model* (for vehicles of high speed) are given. Other mobility models are discussed in [7], and their impact on performance of routing protocols is discussed in [8].

Very little work has been done in maintaining an accurate neighbor set in MANETs. One exception is [9], where a *stable zone* and a *caution zone* of each node have been defined based on a node's position, speed, and direction information obtained from GPS. Specifically, stable zone is the area in which a mobile node can maintain a relatively stable link with its neighbor nodes since they are located close to each other. Caution zone is the area in which a node can maintain an unstable link with its neighbor nodes since they are relatively far from each other. The drawback of this approach is that it is GPS-based, which comes with a cost. In addition, there is no rigorous analysis on the impact of mobility on the selection of these two zones.

Several papers [10] address the issue of the length of time that two nodes will remain close enough in proximity for a link between them to remain active. Several routing protocols, associativity-based routing (ABR) [11] and signal stability-based adaptive routing (SSA) [12], have been proposed that select *stable links* to construct a route. In [13], GPS information is used to estimate the expiration time of the link between two adjacent hosts. Recently, several studies have been done on the effect of mobility on routing path [14]. However, no broadcast protocol uses the notion of stable link to evaluate the stability of neighbor set in order to better decide the forwarding status of each node. Although several probabilistic broadcast protocols [1], [15] have been proposed by trading between efficiency (simple design) and coverage (delivery ratio), it is difficult to establish a direct connection between forwarding probability and node mobility.

B. Broadcast protocols based on self-pruning

Wu and Dai [3] proposed a generic scheme that covers most existing self-pruning protocols. In the generic self-pruning scheme, each node builds its *k-hop information* by exchanging $(k-1)$-hop information with its neighbors via periodical "Hello" messages. Here we define the k-hop neighbor set $N_k(v)$ of node v as the set of nodes that is at most k hops away from v, and the exact k-hop neighbor set $H_k(v)$ as the set of nodes that is exactly k-hops away from v. That is, $N_k(v) = H_1(v) \cup H_2(v) \cup \ldots \cup H_k(v)$. The k-hop information of a node v contains the topology information that can be collected via k rounds of "Hello" message exchanges, including nodes in $N_k(v)$, links among nodes in $N_{k-1}(v)$, and links between $H_{k-1}(v)$ and $H_k(v)$. For example, links between two nodes exactly 2 hops away are included in 3-hop information, but not in 2-hop information. The "Hello" messages also propagate the *priority* of each node, which could be a permanent property (e.g., node id) or a dynamic one (e.g., node degree). During a broadcast process, each node may also extract from the incoming broadcast packets a list of *visited nodes* that have forwarded the same broadcast packet. Using the k-hop topology, priority, and visited node information, each node decides its own status (forwarding/non-forwarding) based on the following coverage condition.

Coverage Condition [3]: Node v has a non-forward node status if for any two neighbors u and w, a *replacement path* exists that connects u and w via several intermediate nodes (if any) with either higher priority values than the priority value of v or with visited node status.

Assume node id is used as priority, node x in Figure 2 (a) is a non-forward node based on the coverage condition, because its neighbors, v and w, are connected via a *replacement path* that contains only intermediate nodes (in this case, y) with higher node id than x, while node y is a forward node, because no such replacement path exists. It was proved in [3] that the coverage condition ensures the coverage; that is, the forward nodes, including the source, form a CDS and, therefore, the delivery of the broadcast packet to every node is guaranteed in a connected network, given that no packet is lost due to node mobility or MAC layer collision.

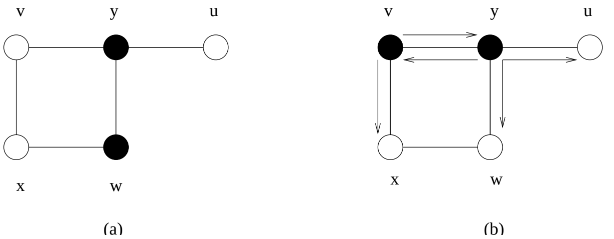


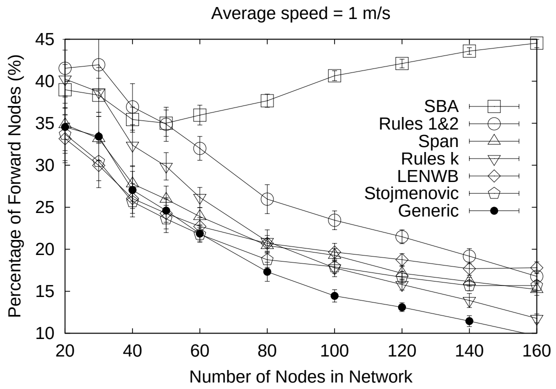
Fig. 2. (a) Forward node set without history information (static). (b) Forward node set with upstream history information (dynamic) with node v being the source (visited node).

A self-pruning protocol is *static* if it does not use visited node in the replacement path; otherwise, it is *dynamic protocol*. In a static protocol, the CDS is constructed before the broadcasting process starts and, hence, is source independent. Dynamic protocols are source dependent and usually have lower broadcast redundancy. For example, node u in Figure 2 (a) is a forward node in a static protocol, as there is no node with higher priority that connects neighbors x and y. When node v issues a broadcasting, the broadcast packet is sent three times by nodes v, w and y. In Figure 2 (b), the forward node status of each node is determined during a broadcast process, and the upstream history information is piggybacked with the broadcast packet. Because nodes v (source) and y are visited nodes, node w can conclude that it can be a non-forward node since two of its neighbors can be connected using node v (a visited node). The broadcast packet is sent twice in the dynamic protocol, one fewer than in the static protocol.

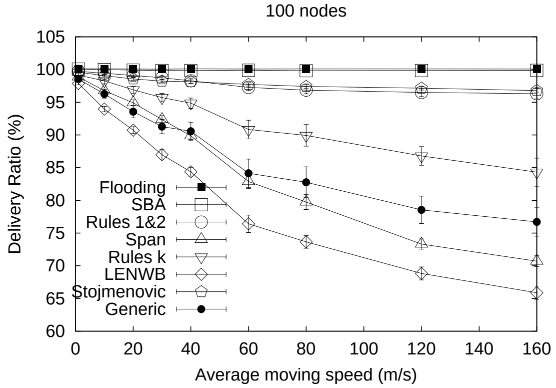
In [3], it is assumed that local views of the broadcast specific information (i.e., visited node information) are dynamic but safe, i.e., an unvisited node will not be mislabelled as visited, and those of the broadcast independent information (i.e., k-hop information and priority) are static and accurate during a broadcast process. However, in mobile networks, the "static" information usually changes and causes inaccurate local views. Based on these inaccurate views, full coverage (i.e., 100% delivery ratio) is not guaranteed. The broadcast redundancy and delivery ratio of a self-pruning protocol in a mobile environment is affected by various implementation options, including:

Priority type: Each node is associated with a priority used to break a tie in replacement. Using node id as priority has higher redundancy than node degree (node id is used then if there is a tie in node degrees) in relatively sparse networks. On the other hand, using node id as priority has higher delivery ratio than node degree in mobile networks. Node id also has less redundancy in dense networks.

"Hello" interval: Using smaller "Hello" interval can provide fresher neighborhood information and improve the delivery ratio in a highly mobile environment. However, small "Hello" intervals can only reduce, but not eliminate, undetected topology changes. Furthermore, if "Hello" interval is too short, the overall broadcasting cost can be higher than flooding (i.e., the



(a) Broadcast redundancy versus network size



(b) Delivery ratio versus mobility

Fig. 3. Performance of various broadcast protocols.

network is flooded with “Hello” messages).

Backoff delay: Dynamic protocols use visited node information to reduce broadcast redundancy. A random backoff delay, the time between the first receipt of the broadcast packet and the forwarding decision, can be used to discover more visited nodes and further increase self-pruning efficiency. In some protocols like SBA [16], using a large backoff delay is essential for the broadcast efficiency. However, a large backoff delay also causes large end-to-end delay. A random jitter delay is also used by each node to avoid collision, but is usually too short to affect the broadcast redundancy or delivery ratio.

Location information: A protocol using location information obtained from a GPS device has smaller “Hello” messages and fresher neighborhood information than other protocols [17]. On the other hand, GPS devices cause extra cost and energy consumption. Location information obtained may be inaccurate. In addition, neighbor set based on distance (from GPS) may not be reliable, since it is well known that the time variation of the channel strength can be due to many other factors including multipath fading, shadowing by obstacles, and interference from other users.

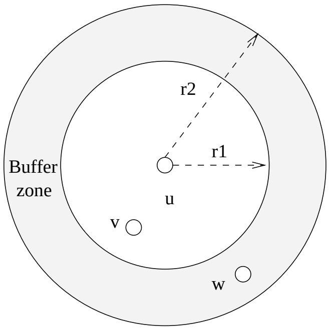


Fig. 4. Forward node selection and forwarding process based on two different transmission ranges: r_1 and r_2.

Six existing algorithms, including static and dynamic protocols, were shown to be special cases of the coverage condition. They are: Wu and Li’s marking process with Rules 1 & 2 (static) [18], Dai and Wu’s Rule k (static) [19], Chen et al’s Span (static) [20], Sucec and Marsic’s LENWB (dynamic) [21], Peng and Lu’s SBA (dynamic) [16], and Stojmenovic’s algorithm (hybrid) [17]. Details of these algorithms are given in Appendix.

As shown in Figure 3, high delivery ratio can be achieved by protocols with high broadcast redundancy, i.e., blind flooding, SBA, and Rules 1&2. The new protocol (labeled as Generic) has the lowest redundancy, but suffers from low delivery ratio in highly mobile networks. One solution is to use location information as in Stojmenovic’s algorithm, which achieves higher delivery ratio with relatively low redundancy. However, using location information incurs extra cost and may not provide accurate prediction on the existence of wireless links. SBA achieves very high delivery ratio in highly mobile networks, but it also has the highest percentage of forward nodes. Note that the percentage of forward nodes in SBA is a function of its backoff delay. In networks with relatively low mobility, a longer backoff delay can be used to improve the efficiency of SBA. However, this also incurs longer end-to-end delay, which is undesirable under certain circumstances, e.g., in route discovery and in applications with highly mobile nodes.

III. PROPOSED METHOD

This section proposes a mobility control method that addresses connectivity, link availability, and consistency issues. Two sufficient conditions, one on the connectivity of the physical network that ensures connectivity of the virtual network and the other on the bound of the range difference that ensures link availability, are given. Then we introduce methods to relax these sufficient conditions based on probabilistic analysis and optimization techniques.

A. Basic Idea

We propose a mobility management method without resorting to location information. This approach is based on two transmission ranges, r_1 and r_2, with $r_1 < r_2$. r_1 is used to collect neighbor set and k-hop information through "Hello" messages, whereas r_2 is used to perform actual transmission. Specifically, the proposed method consists of two stages: (a) forward node selection, followed by (b) forwarding process. Assume the first stage is done dynamically during the broadcast process.

- **Forward node selection**: Select a small forward node set using an existing method where each neighbor set is based on transmission range r_1.
- **Forwarding process**: Whenever a node receives a message for the first time, if it is a forward node, it forwards the message using transmission range r_2.

A node that is within the range of r_1 of node u is called a neighbor of u and the collection of such nodes is the neighbor set of u. The set of nodes that are reachable based on r_2 is called *effective neighbor set*. Figure 4 shows the relationship between these two transmission ranges. In this example, v is in u's neighbor set (also in u's effective neighbor set), whereas w is in u's effective neighbor set (but not in u's neighbor set).

The idea of two transmission ranges is to use the "ring", the area bounded by two circles with transmission ranges r_1 and r_2, as a buffer zone to nullify the various bad effects caused by node mobility and transmission delay. However, one bad effect called inconsistent local views cannot be nullified no matter how wide the buffer zone is. Inconsistent local views ultimately result in "bad decision" from a node. A decision is *bad* if a node that should forward the message decides on a non-forwarding status.

B. Physical and logical networks and broadcast states

In [3], the coverage condition was applied on a static or semi-static physical network. That is, the physical topology stops to change several "Hello" intervals before a broadcast process, and stays unchanged until the broadcast process completes. For the sake of clarity, we assume node id is used as priority, and define the local view of each node as a subgraph of the physical topology (i.e, k-hop information). The correctness of the coverage condition is based on the assumption that every node decides its forwarding/non-forwarding status based on a "fresh" view. In MANETs, however, this assumption can be easily violated due to the continuous mobility. In fact, in order to apply the coverage condition on MANETs with potentially obsolete local views, we introduce the concepts of *logical network* and *broadcast state*. As shown in Figure 5. A logical network is the collection of all local views, i.e., a super graph containing all the nodes and links in local views. Note that the logical network is dynamic in a MANET. When the physical topology changes, the change is detected by "Hello" messages and reflected in the logical network. Broadcast state, defined as follows, is a snapshot of local views. For a specific broadcast process, broadcast state forms a virtual static network, upon which the coverage condition is applied.

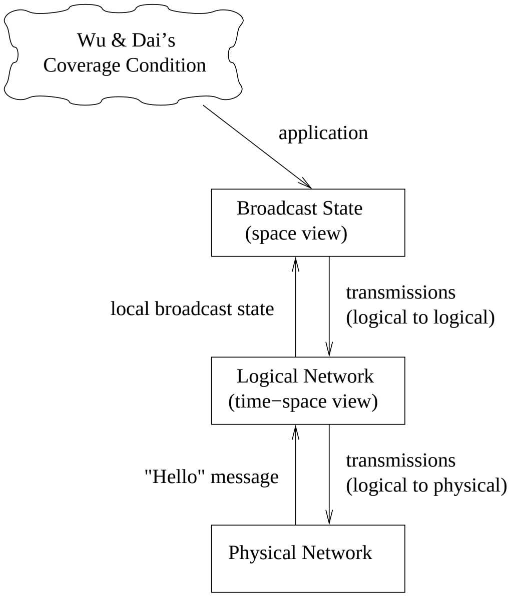


Fig. 5. The mapping from the logical network and broadcast state to the physical network.

Definition 1: A *local broadcast state* for a broadcast is a local view at the time the forwarding/non-forwarding decision is made at an individual node. A *(global) broadcast state* is the collection of all local broadcast states for a specific broadcast.

We assume that each node has the same "Hello" interval f[3], but each node starts its period asynchronously. In order to build k-hop information, each node advertises its $(k-1)$-hop information via "Hello" messages. Each node updates its local view based on received "Hello" messages. Because of asynchronous periodic exchanges among neighboring nodes, the 1-hop neighbor set in a local view at a particular time t does not reflect the actual neighbor set at time t, but the offset is bounded by the "Hello" interval f. In fact, k-hop information is a set that consists of neighborhood information sampled at different times. In general, $H_{i+1}(u)$ was sampled one interval after $H_i(u)$ for $i = 1, 2, ..., k-1$. Clearly, the k-hop information at time t does not reflect the actual neighborhood topology at time t, and the offset is bounded by kf. Suppose the speed of node movement is upper bounded by s. Then sf is the maximum distance a node can move around during a "Hello" interval. The *maximum relative distance* between two nodes in such an interval is $\Delta = 2sf$.

Consider the MANET in Figure 2 (a) and a broadcast process, which is first from v to x, y, and then from y to u, w, Figure 6 (a) shows the update of local views. We label

---

[3] The condition can also be be relaxed in a controllable way, such as $(1 \pm 0.25)f$ in AODV.

the time each node sends its last “Hello” message before the broadcasting as t_i, and the time for previous “Hello” messages as t_{i-1}, t_{i-2}, and so on. Note that t_i at each node may refer to different physical time. Here each node builds 2-hop information. If node y’s “Hello” message is first received by node v between t_{i-2} and t_{i-1} (the “Hello” message propagation is shown in a dotted arrow line), it is added to v’s 1-hop neighbor set, which is advertised in v’s next “Hello” message at t_{i-1}. That is, link (v, y) is added to local views of nodes v and x. Similarly, link (w, y) is also detected and added to local views of nodes w and x.

Recall in self-pruning, each node follows three steps: (a) first receipt of broadcast message, (b) backoff delay, and (c) forward/non-forward status decision and transmission (if needed). A *broadcast period* starts from the source sending out the message and ends with the last node deciding its forwarding status. Like [3], it is assumed that the broadcast message propagates quickly and its delay can be ignored. Backoff at intermediate nodes are allowed, but *accumulative backoff* along each path of the broadcast tree is bounded by b, called *broadcast delay*, for each broadcast. Note that b may also include broadcast message propagation delay if such delay cannot be neglected. In Figure 6 (a), the time that each individual node makes its decision is marked with a black dot. Note that local broadcast states are taken at the times marked by these black dots, and the global broadcast state is the collection of local broadcast states (marked by the dashed line connecting all black dots).

C. Proposed Methods

Wu and Dai’s coverage conditions can be applied to the global broadcast state and ensures coverage, given that the following three conditions are met:

Connectivity: The virtual network that corresponds to the global broadcast state should be connected in order to apply Wu and Dai’s condition. The following theorem shows the density requirement at the physical network for ensuring a connected virtual network.

Theorem 1: If the physical network with transmission range $r_1 - \Delta'$ is connected under all time, where $\Delta' = 2s(f + b)$, then every virtual network induced from a global broadcast state is connected.

Proof: Assume the global broadcast state is taken in a broadcast process started at time t. Since the maximum broadcast delay is b, all local states are taken within time period $[t, t + b]$. If the distance of two nodes u and v, $d(u, v) \leq r_1 - 2s(f + b)$ at time $t - f$, then $d(u, v) \leq r_1$ during $[t-f, t+b]$. Suppose u takes its local broadcast state at $t_u \in [t, t+b]$, it must have received at v’s last “Hello” message in $[t-f, t_u]$. Therefore, link (u, v) exists in u’s local broadcast state. Since the global broadcast state consists of all links from local broadcast state, and the network is connected at time $t-f$ in the range of $r_1 - \Delta'$, the corresponding virtual network induced from the global broadcast state is also connected. ■

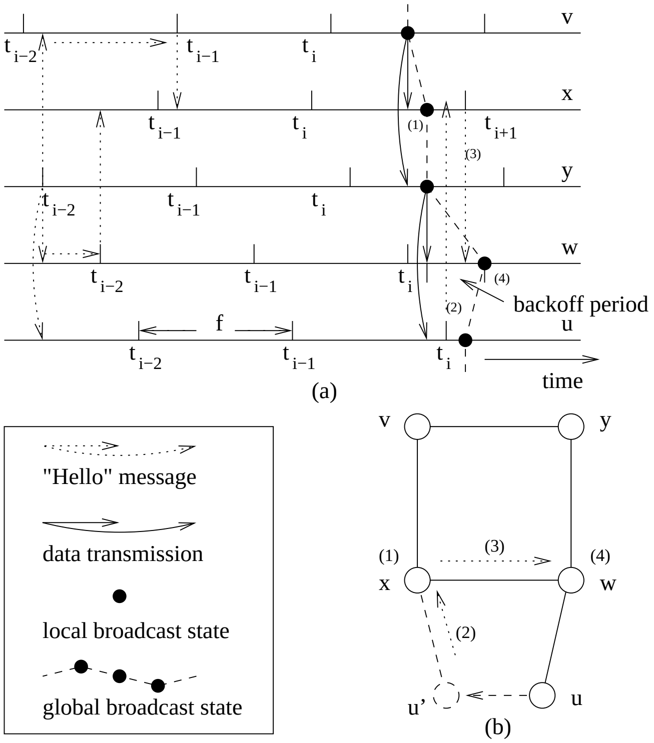


Fig. 6. The time-space view of the logical network of Figure 2 (a) space view of an inconsistent global broadcast state, after node u is identified by node x as its new neighbor (b).

Theorem 1 poses a rather strict connectivity requirement on the physical network. That is, if the physical network cannot meet the connectivity requirement, the virtual network is not guaranteed to be connected and Wu and Dai’s approach will fail. We will discuss later an approach that relaxes the connectivity requirement under the cost of pruning efficiency.

Link availability: Any link in the global broadcast state should still exist in the physical network during the broadcast period (i.e., a neighbor sampled with range r_1 is still a neighbor in the range of r_2 during the broadcast period).

Theorem 2: To ensure the link availability requirement, r_2 should be set so that $\Delta'' \leq r_2 - r_1$, where $\Delta'' = k\Delta + \Delta'$ and k for k-hop information.

Proof: (sketch) We need to show that any neighbor under the transmission range r_1 when its state is sampled is still an effective neighbor under the transmission range r_2 when the message is sent out. The total delay includes k-hop neighbor set collection that takes k intervals, and $(f + b)$ broadcast and synchronization delay. The former contributes a distance of $k\Delta$ and the latter Δ'. ■

The above analysis provides some theoretical foundations for ensuring full coverage. However, the analysis shows only the worst case situation, which rarely occurs. Later we will show that even when $r_2 - r_1$ is much smaller than Δ'', the probability of a undetected link failure is very low. Since most

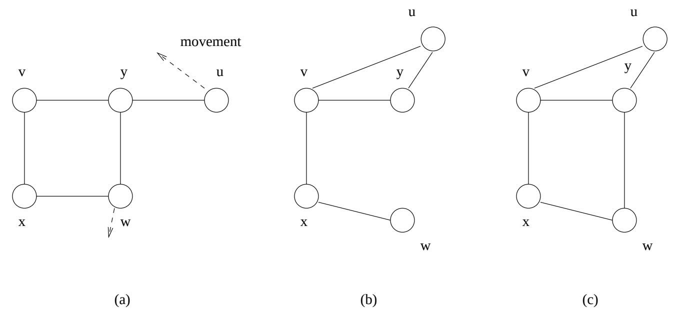
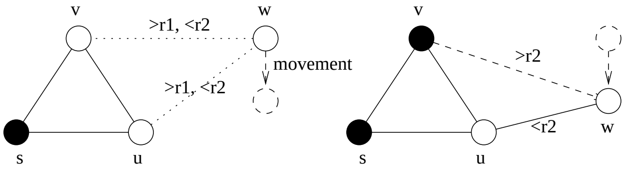


Fig. 7. The physical network changes from (a) to (b). (c) collection of aggregated local states.



(a) "Hello" range r1 causes partition

(b) "Hello" range r2 causes link failure

Fig. 8. A network with one mobile node w (a) before the movement and (b) after the movement. Dotted lines represent undetected physical links. The dashed line represents a undetected broken link.

self-pruning protocols have certain degrees of redundancy, it usually takes several undetected link failures to fail a broadcast. That is, the probability is high that full coverage can be achieved with a relatively small buffer zone width. There is a wide range of potential tradeoffs between broadcast efficiency and broadcast delivery ratio.

Consistency: Two local views of nodes u and v are *inconsistent*, if there exits a link (v, w) in u's k-hop information, but v does not view w as a 1-hop neighbor. For example, assume the physical topology in Figure 2 changes shortly before the broadcast. The broadcast may fail due to inconsistent views. Figure 7 (a) shows the physical network before the change, where node x is a non-forward node because its neighbors v and w are connected via a replacement path (v, y, w). Figure 7 (b) shows the physical network before the broadcast, where y is a non-forward node because w is no longer a neighbor, and the remaining two neighbors v and u are directly connected. Node y detects the broken link (y, w) before node x, since y is adjacent to the link while x is 2-hop away from the link. Both nodes may take a non-forwarding status in the broadcast, x's decision based on the outdated view and y's based on the updated view. Therefore, node w may never receive the broadcast packet.

We propose to use the *aggregated local state* to address the inconsistency problem. The main problem of the above example is that node y removes link (y, w) in its local view before node x does so. Note that any broken link is detected first as the loss of a 1-hop neighbor by the end nodes. This link is not removed from local views of other nodes until the link failure is advertised via "Hello" messages. When k-hop information is used, it takes up to k "Hello" intervals for all related nodes to update their local views. The solution is that once *a node advertises its 1-hop neighbor set, it cannot back away from it immediately*. That is, each node v keeps k recent versions of $N(v)$ advertised in its last k "Hello" messages. The local state used to make the forwarding/non-forwarding decision in a broadcast is the *aggregation* of the k advertised local views. The aggregation takes 1-hop neighbors from all k views, but other information from the last view only. The rationale is that node u still views node w as its 1-hop neighbor until link (u, w) is removed from local views of all nodes in $N_k(u) \cap N_k(w)$. Figure 7 (c) shows the collection of aggregated local views. In this case, node y will still forward. Intuitively, once a node v appears as a neighbor of u (in the range of r_1) during the recent k intervals, it still has to be treated as a neighbor even if it currently moves out of u's visible range, but is still in u's effective neighbor set (as shown in Theorem 2).

Another form of inconsistency might occur if a node w uses the "Hello" message from x sent after x made its decision (forwarding/non-forwarding). As shown in Figure 6 (b), node u is initially a neighbor of w and later moves to x as its neighbor. If "Hello" message is sent from x to w after x has made its decision, but before w made its, then w's decision is made based upon information that is not available to x when it made its decision. Consider the following sequence of events as shown in the Figures 6 (a) and (b): (1) x decides its non-forwarding status, (2) u is detected by x as a new neighbor, (3) x advertises its new neighbor set, and (4) w believes that u is covered by x and becomes a non-forward node. In this case, u will never receive the broadcast packet. A simple solution is for each node that has made a decision on a broadcasting to piggyback broadcast id (which is a tuple of source id and sequence number) and timestamp (the time the decision is made) to the "Hello" message. The receiver can then ignore the "Hello" message of a sender sent after the decision is made at the sender. Note that the broadcast period is bounded by b; only recent broadcast id's within b need to be piggybacked into the "Hello" message.

D. Implementation Details

According to Theorem 1, full coverage is guaranteed only when the network is dense enough. In the following, we propose a mechanism that relaxes the connectivity requirement under the cost of pruning efficiency. In sparse networks, using a small "Hello" transmission range may cause partition in the logical network. As shown in Figure 8 (a), when the "Hello" transmission range is r_1, neither node u nor v view node w as a neighbor, because they cannot receive "Hello" messages from w. Therefore, both u and v become non-forward nodes, and node w will not receive the broadcast packet. Simply increasing the "Hello" transmission range to r_2 cannot solve the problem. Since there is no more "buffer zone" that tolerates node before a topology change is detected and propagated to the neighborhood. As shown in Figure 8 (b), u becomes a non-forward node, relying on v to forward the packet w. Meanwhile, node w moves out of the transmission range of v and will not receive the packet from v either. Here we have a dilemma on the maximal distance between two neighbors in the logical network. If two nodes are viewed as neighbors only when their distance is less than r_1, the broadcast may fail due to partition. If two nodes with distance larger than r_1 are viewed as neighbors, the broadcast may fail due to the lack of buffer zone.

Our solution is based on maintaining two neighbor sets. The *covered neighbor set*, $N_c(v)$, of node v consists of all nodes within the normal (large) transmission r_2, and the *advertised neighbor set*, $N_a(v)$, consists of only nodes with distance less than r_1. If v is a non-forward node, every pair of nodes in $N_c(v)$ must be connected via a replacement path. In this case, node v in Figure 8 (a) views w as a neighbor and becomes a forward node. On the other hand, only $N_a(v)$ is propagated to neighbors to build their k-hop information. Therefore, link (v, w) in Figure 8 (b) is invisible to node u. Node u also forwards the broadcast packet and ensures the coverage. Note that this method is conservative. If link (v, w) is still available, making node u a forward node causes extra redundancy.

The dual neighbor sets are constructed via using two "Hello" transmission ranges: the normal transmission range r_2 and the reduced transmission range r_1. This mechanism can be further improved, if each node can estimate its distance to a neighbor based on "Hello" signal strength. In this case, "Hello" messages are sent via the normal transmission range r_2. Each node constructs its covered and advertised neighbor sets based on the estimated distances.

E. Analytical Study

Based on Theorem 2, in order to guarantee that a neighbor (within r_1) at t_0 is an effective neighbor (within r_2) at a time $t_1 = t_0 + f$, r_1 must be smaller than $r_2 - 2sf$ for a given maximal node speed s and time period f. In this section, we show that the probability, p, that a node within r_1 at t_0 moves out of range r_2 at t_1 is reasonably small with a much larger r_1. We assume a mobility model similar to the random direction model [22], where each node is moving at a random speed in $[0, s]$ to a random direction in $[0, 2\pi]$. This is a simplified model for ease of probabilistic analysis. In addition, this model usually represents the worst case in terms of relative distance between two nodes in a given interval.

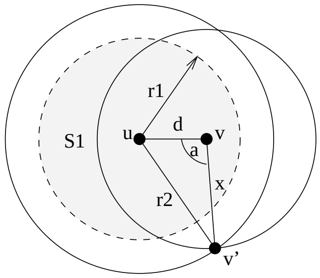


Fig. 9. Calculation of the probability that a neighbor within the "Hello" transmission range (r_1) moves out of the normal transmission range (r_2).

Consider two neighboring nodes u and v (as shown in Figure 9). Node v is within u's "Hello" transmission range (the shadowed area) at time t_0, and moves to position v' at t_1. Assume that their distance at t_0 is d, and v moves a distance of x with respect to u at t_1. The probability that v moves out of the normal transmission range of u is

$$p(x, d) = \begin{cases} 0 & : \ x < r_2 - d \\ 1 - \frac{\alpha}{\pi} & : \ r_2 - d \le x \le r_2 + d \\ 1 & : \ x > r_2 + d \end{cases} \tag{1}$$

where

$$\alpha = \cos^{-1}\left(\frac{x^2 + d^2 - r_2^2}{2dx}\right)$$

is the largest value of $\angle uvv'$ that satisfies $d(u, v') \le r_2$. The probability that *any* node within the "Hello" transmission range of u moves out of its normal transmission range at t_1 is

$$p(x) = \int_0^{r_1} \frac{2\pi d}{S_1} p(x, d)\,\mathrm{d}d = \int_0^{r_1} \frac{2d}{r_1^2} p(x, d)\,\mathrm{d}d \tag{2}$$

where

$$S_1 = \pi r_1^2$$

is the area within the "Hello" transmission range. The probability that a node with *any* constant relative speed with respect to u moves out of the normal transmission range is

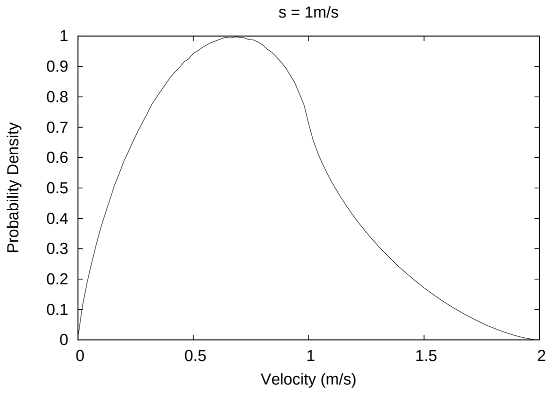
$$p = \int_0^{2s} f_{|\vec{V}|}(y)\, p(fy)\,\mathrm{d}y \tag{3}$$

Here $\vec{V} = \vec{V_v} - \vec{V_u}$ is the random joint mobility vector between any two mobile nodes u and v, where $\vec{V_u}$ ($\vec{V_v}$) is the random mobility vector of node u (v). Note that equation (1) still holds, as the direction of $\vec{V}$ is also uniformly distributed in $[0, 2\pi]$, and is independent of the speed of $\vec{V}$, $|\vec{V}|$. We know that $|\vec{V}|$ is between 0 and $2s$; $|\vec{V}| = 0$ when $\vec{V_u} = \vec{V_v}$, and $|\vec{V}| = 0$ when $\vec{V_u} = -\vec{V_v}$ and $|\vec{V_u}| = |\vec{V_v}| = s$. However, its probability function, $f_{|\vec{V}|}(t)$, is unknown. McDonald and Znati [23] conducted a probabilistic analysis on the joint mobility of two nodes, but their analysis is based on the random walk mobility model [6], where the mobility vector of each node is the sum of several epochs, each epoch has different speed, direction, and duration. Li, Hou and Sha's analysis [24] is based on the same mobility model as ours, but their analysis is simplified by the implicit assumption that node u is fixed and $|\vec{V}|$ is uniformly distributed in $[0, s]$. Here we calculate $f_{|\vec{V}|}(t)$ at a given t as

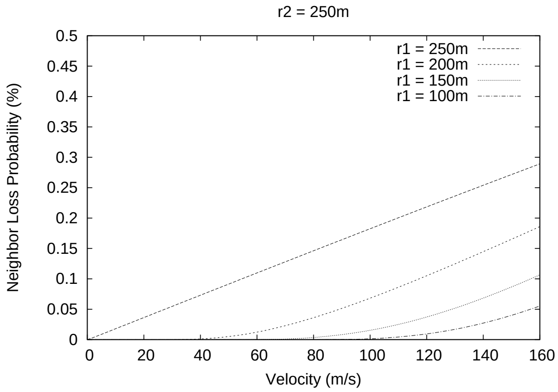
$$\begin{aligned} f_{|\vec{V}|}(t) &\approx \frac{F_{|\vec{V}|}(t + \delta t) - F_{|\vec{V}|}(t)}{\delta t} \\ &= \frac{P(t \le |\vec{V}| \le t + \delta t)}{\delta t} \\ &= \oint_{(0,0)}^{(2\pi,s)} \oint_{(0,0)}^{(2\pi,s)} \frac{R(\vec{V_u}, \vec{V_v}, t, t + \delta t)}{(2\pi s)^2 \delta t} \mathrm{d}\vec{V_u} \mathrm{d}\vec{V_v} \end{aligned} \tag{4}$$

where $F_{|\vec{V}|}(t)$ is the distribution function, δt is a small positive value, and

$$R(\vec{V_u}, \vec{V_v}, a, b) = \begin{cases} 1 & : \ a \le |\vec{V_v} - \vec{V_u}| \le b \\ 0 & : \ \text{otherwise} \end{cases}$$



(a) The probability function $f_{|\vec{V}|}(t)$ of the random joint mobility vector.



(b) The probability that a neighbor within a given "Hello" transmission range r_1 moves out of the normal transmission range r_2.

Fig. 10. Calculation results.

Figure 10 (a) shows the distribution of $|\vec{V}|$ calculated from (4), when $s = 1m/s$ and $\delta t = 0.001m/s$. Note that the probability that $|\vec{V}| > 1.5s$ is small ($\leq 5\%$). Based on this distribution, we calculate the probability p that any node within the "Hello" transmission range ($r_1 = 100$, 150, 200, and 250) of u moves out of its normal transmission range ($r_2 = 250m$) during a "Hello" interval ($f = 1s$), when the maximal single node speed s varies from 0 to $160m/s$. As shown in Figure 10 (b), we can use an r_1 that is much larger than $r_2 - 2sf$, and still expect a low probability that an effective neighbor moves out of the normal transmission range. For example, when $r_1 = 200m$ and $s = 80m/s$, the probability of losing an effective neighbor is less than 5%. Note that the corresponding r_1 that guarantees the availability of link (u, v) at time t_1 is $r_2 - 2sf = 90m$. When $r_1 = 100m$ and $s = 160m/s$, the probability of losing an effective neighbor is about the same. On the other hand, there is no r_1 that can guarantee the link availability, as $2sf = 320m > r_2$.

TABLE I

SIMULATION PARAMETERS

Parameter	**Value**
Network area	$900 \times 900\ m^2$
Number of nodes	50, 100
Average moving speed	1-160 m/s
Pause Time	0 s
Normal transmission range	250 m
"Hello" transmission range	100-250 m
"Hello" interval	0.75-1.25 s
Priority type	node id
Backoff delay	N/A
Location information	N/A
Simulation time	100 s
Number of trials	20
Confidence level	95%

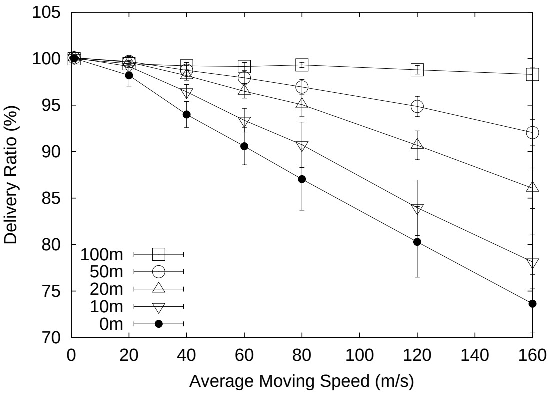
IV. SIMULATION

Simulations are conducted to evaluate the proposed method and explore appropriate "Hello" transmission ranges that achieve high delivery ratio with low broadcast redundancy under various mobility levels. We also evaluate the effectiveness of two implementation options that use dual neighbor sets to improve the delivery ratio under various environments.

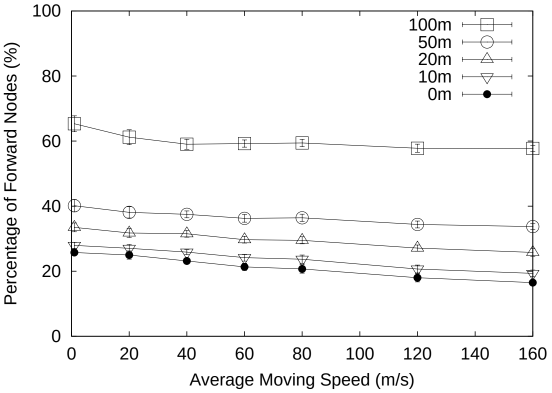
A. *Simulation environment*

The proposed mobility management method is simulated on *ns-2*($1b7a$) [25] and its CMU wireless extension. We extend the Wu and Dai's coverage condition by using two transmission ranges r_1 (for "Hello" messages) and r_2 (for actual transmission). When $r_1 = r_2$, the new algorithm is equivalent to the original generic self-pruning protocol. We also simulate the dual neighbor sets enhancement for sparse networks. The configuration of mobile networks and the implementation parameters of the extended coverage condition are listed in Table I. Since our purpose is to observe the behavior of self-pruning protocols under mobile environments, all simulations use an ideal MAC layer without contention or collision. If a node sends a packet, all neighbors within its transmission range will receive this packet after a short propagation delay. We assume that accurate location information is either unavailable, or unable to predict the existence of wireless links due to the irregular variation of transmission range. It was shown in [2] that the contribution of a backoff delay to the protocol efficiency is trivial except for SBA. Therefore, our implementation of the proposed method does not use a backoff delay.

The mobility model used in the simulation is the random direction model [22]. In this model, each node heads in a random direction and moves at a random speed until it reaches the boundary of the area, where it selects new direction and speed and keeps moving. Our mobility pattern generator is from [6], which has a parameter called average moving speed (V_{avg}). For a given V_{avg}, the speed of each node is randomly selected from the range $[0, 2V_{avg}]$. Note that the random direction model usually yields sparser networks and higher mobility than the commonly used random waypoint model [6]. Therefore, a reliable protocol in this simulation study is

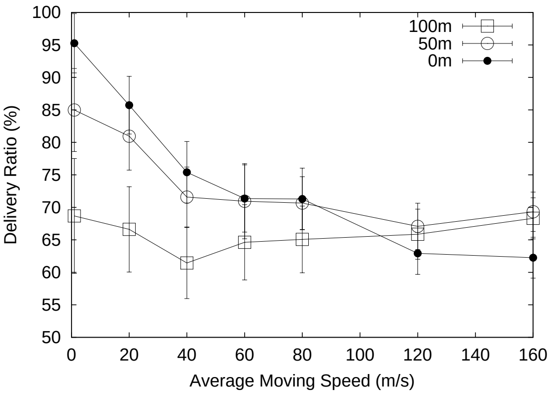


(a) Delivery ratio.

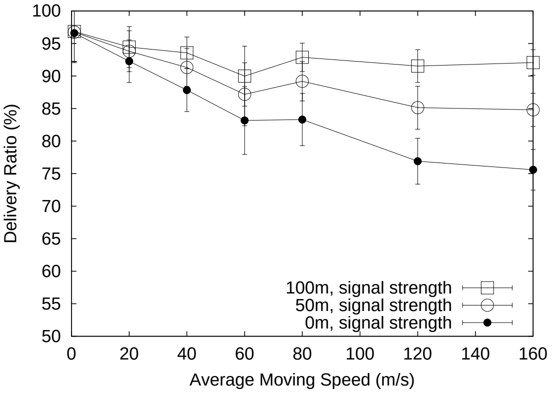


(b) Broadcast redundancy.

Fig. 11. Simulations in relatively dense (100 nodes) networks.



(a) The original single neighbor set method.



(b) The dual neighbor set enhancement.

Fig. 12. Delivery ratio in relatively sparse (50 nodes) networks.

a reliable protocol under the random waypoint model, but not vice versa.

B. Simulation results

Figure 11 shows simulation results in relatively dense networks (100 nodes), with buffer zone width (i.e., $r_2 - r_1$) varying from $0m$ to $100m$. As expected, high delivery ratio ($\geq 98\%$) can be achieved with large buffer zone width ($100m$) in highly mobile networks (with average speed $160m/s$). The only problem is the high broadcast redundancy ($\geq 60\%$ forward nodes). If the network mobility level is known, we can select the buffer zone width based on the mobility level to balance the delivery ratio and redundancy. For example, at average speed $120m/s$, we can use a buffer zone width of $50m$, which achieves 95% delivery ratio with 40% forward nodes. At average speed $40m/s$, a $10m$ buffer zone achieves the same delivery ratio with only 30% forward nodes.

Figure 12 (a) shows the delivery ratio of the proposed method in relatively sparse networks (50 nodes). When a $0m$ buffer zone is used, the delivery drops rapidly as the average speed increases. Using a larger buffer zone width ($50m$ or $100m$) improves the delivery ratio under high mobility level, but performs poorly under low mobility level. The delivery ratio is low (85% and 70%), even with trivial mobility ($1m/s$). One reason for the low delivery ratio in sparse networks is the relatively low redundancy. Simulation results in [2] showed that all self-pruning protocols have lower delivery ratio in sparse networks than in dense networks under the same mobility level. Another reason is that when the network is not dense enough, the connectivity requirement in Theorem 1 is not satisfied, and therefore, cannot guarantee the coverage.

This problem can be solved with dual neighbor set enhancement introduced in subsection III-D. Figure 12 (b) shows the delivery ratio of the enhanced scheme, where all neighbors within the normal transmission range r_2 are put into the covered neighbor set, and only neighbors within the reduced transmission range r_1 are put into the advertised neighbor set. With this enhancement, high delivery ratio ($\geq 90\%$) can still be achieved under the highest mobility level.

Overall, Simulation results show that balance between delivery ratio and broadcast redundancy can be achieved by adjusting the buffer zone width based on the network mobility level. As predicted by our probabilistic analysis, for each mobility level, high delivery ratio can be achieved with a buffer zone much thinner than required by Theorem 2. The dual neighbor set enhancement is proved successful in relaxing the connectivity requirement in Theorem 1, and achieves high delivery ratio in sparse networks.

V. CONCLUSIONS

In this paper, we have proposed a mobility management method based on the use of two transmission ranges. Using this mechanism, we have also extended Wu and Dai's coverage condition to a dynamic environment where network topology is allowed to change, even during the broadcast process. In addition, connectivity, link availability, and consistency issues related to neighborhood information of different nodes have also been addressed. The proposed scheme can also be extended to provide mobility management for other activities such as topology control in MANETs [26].

The constraint used on $r_2 - r_1$ in this paper is conservative. Our probabilistic analysis suggests that high delivery ratio can still be achieved with a larger r_1. Simulation results show that the proposed method and two enhancements achieve good balance between delivery ratio and broadcast redundancy by adjusting the value of r_1 based on the network mobility level.

In Wu and Dai's coverage condition, node id is used to break a tie. We could also use the notion of *relative mobility* [27], defined as absolute relative speed averaged over time, for tie breaking. In general, a node with high relative mobility is more prone to unstable behavior than a node with less relative mobility and therefore should be pruned (from being a forward node) when possible. In this case, relative mobility is calculated locally through some form of approximation and distributed through piggybacking with regular "Hello" messages.

REFERENCES

[1] Y.-C. Tseng, S.-Y. Ni, and E.-Y. Shih, "Adaptive approaches to relieving broadcast storms in a wireless multihop mobile ad hoc network," *IEEE Transactions on Computers*, vol. 52, no. 5, pp. 545–557, May 2003.

[2] F. Dai and J. Wu, "Performance comparison of broadcast protocols for ad hoc networks based on self-pruning," 2003, submitted for publication.

[3] J. Wu and F. Dai, "Broadcasting in ad hoc networks based on self-pruning," in *Proc. of IEEE Infocom*, Mar./Apr. 2003.

[4] M. Grossglauser and D. Tse, "Mobility increases the capacity of ad-hoc wireless networks," in *Proc. of IEEE Infocom*, Apr. 2001.

[5] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388404, Mar. 2000.

[6] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communication & Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications*, vol. 2, no. 5, pp. 483–502, 2002.

[7] C. Schindelhauer, T. Lukovszki, S. Rhrup, and K. Volbert, "Worst case mobility in ad hoc networks," in *Proc. of the 15th annual ACM symposium on Parallel algorithms and architectures*, Aug. 2003.

[8] F. Bai, N. Sadagopan, and A. Helmy, "IMPORTANT: a framework to systematically analyze the impact of mobility on performance of routing protocols for ad-hoc networks," in *Proc. of IEEE Infocom*, Mar./Apr. 2003.

[9] W.-I. Kim, D.-H. Kwon, and Y.-J. suh, "A reliable route selection algorithm using global positioning systems in mobile ad-hoc networks," in *Proc. of ICC*, June 2001.

[10] S. Basagni, I. Chlamtac, A. Farago, V. R. Syrotiuk, and R. Talebi, "Route selection in mobile multimedia ad hoc networks," in *Proc. of IEEE MOMUC*, Nov. 1999.

[11] C.-K. Toh, "Associativity-based routing for ad-hoc mobile networks," *IEEE Personal communications*, vol. 4, no. 2, pp. 36–45, Mar. 1997.

[12] R. Dube, C. Rais, K. Wang, and S. Tripathi, "Signal stability based adaptive routing (SSA) for ad hoc mobile networks," *IEEE Personal Communications*, vol. 4, no. 1, pp. 36–45, Feb. 1997.

[13] W. Su, S.-J. Lee, and M. Gerla, "Mobility prediction and routing in ad hoc wireless networks," *International Journal of Network Management*, vol. 11, no. 1, pp. 3–30, Jan.-Feb. 2001.

[14] N. Sadagopan, F. Bai, B. Krishnamachari, and A. Helmy, "Paths: analysis of path duration statistics and their impact on reactive MANET routing protocols," in *Proc. of MobiHoc*, June 2003.

[15] Z. J. Haas, J. Y. Halpern, and L. Li, "Gossip-based ad hoc routing," in *Proc. of IEEE Infocom*, June 2002.

[16] W. Peng and X. Lu, "On the reduction of broadcast redundancy in mobile ad hoc networks," in *Proc. of MobiHoc*, 2000, pp. 129–130.

[17] 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.

[18] J. Wu and H. Li, "On calculating connected dominating set for efficient routing in ad hoc wireless networks," in *Proc. of DiaLM*, 1999, pp. 7–14.

[19] F. Dai and J. Wu, "Distributed dominant pruning in ad hoc wireless networks," in *Proc. of ICC*, May 2003, vol. 1, pp. 353–357.

[20] B. Chen, K. Jamieson, H. Balakrishnan, and R. Morris, "Span: An energy-efficient coordination algorithm for topology maintenance in ad hoc wireless networks," in *Proc. of MobiCom*, July 2001, pp. 85–96.

[21] J. Sucec and I. Marsic, "An efficient distributed network-wide broadcast algorithm for mobile ad hoc networks," CAIP Technical Report 248, Rutgers University, Sep. 2000.

[22] E. Royer, P. Melliar-Smith, and L. Moser, "An analysis of the optimum node density for ad hoc mobile networks," in *Proc. of ICC*, 2001.

[23] A. B. McDonald and T. F. Znati, "A mobility-based framework for adaptive clustering in wireless ad hoc networks," *IEEE JSAC, Special Issue on Ad-Hoc Networks*, Aug. 1999.

[24] N. Li, J. C. Hou, and L. Sha, "Design and analysis of an MST-based topology control algorithm," in *Proc. IEEE Infocom*, Mar./Apr. 2003.

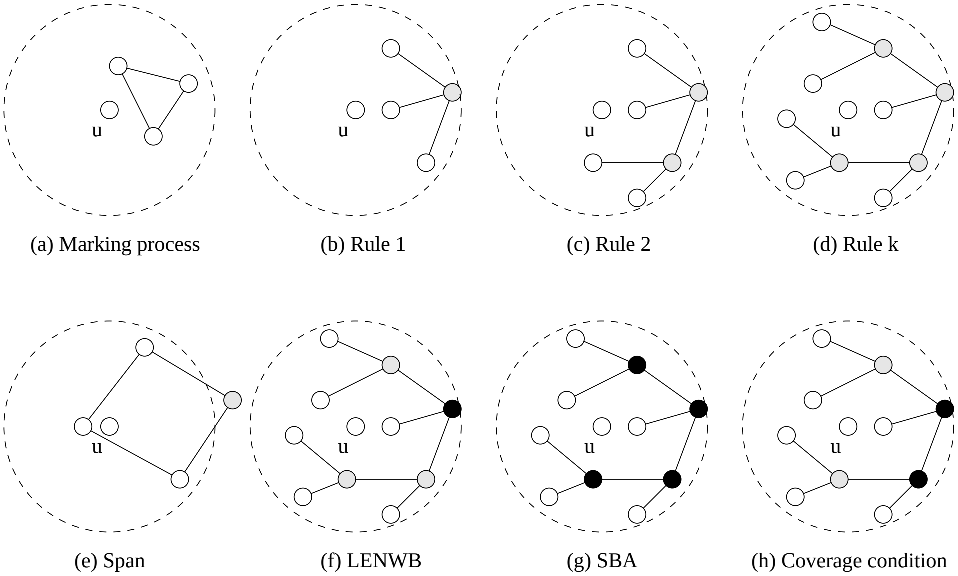
[25] K. Fall and K. Varadhan, "The *ns* manual," The VINT Project, UCB, LBL, USC/ISI and Xerox PARC, http://www.isi.edu/nsnam/ns/doc/, Apr. 2002.

[26] J. Wu and F. Dai, "Mobility-sensitive topology control in mobile ad hoc networks," Oct. 2003, submitted for publication.

[27] B. An and S. Papavassiliou, "A mobility-based clustering approach to support mobility management and multicast routing in mobile ad-hoc wireless networks," *International Journal of Network Management*, vol. 11, no. 6, pp. 387–395, Nov.-Dec. 2001.

APPENDIX: SPECIAL CASES OF THE COVERAGE CONDITION

Wu and Li's algorithm (static): Wu and Li [18] proposed a *marking process* to determine a set of *gateways* (i.e., forward nodes) that form a CDS: a node is marked as a gateway if it has two neighbors that are not directly connected. Two pruning rules are used to reduce the size of the resultant CDS. According to pruning Rule 1, a gateway u can become a non-gateway if all of its neighbors are also neighbors of another node v that has higher priority value; that is, u's neighbor set is *covered* by v. According to pruning Rule 2, a marked node can be unmarked if all of its neighbor set is covered by two other nodes that are directly connected and have higher priority values.



(a) Marking process (b) Rule 1 (c) Rule 2 (d) Rule k

(e) Span (f) LENWB (g) SBA (h) Coverage condition

Fig. 13. Node u in the center of each subgraph can be self-pruned by the corresponding protocol. Nodes in the transmission range of node u (the dashed circle) are neighbors of u. Gray nodes have higher priorities (e.g., higher id's) than u. Black nodes are visited nodes that have forwarded the broadcast packet.

Dai and Wu's algorithm (static): Dai and Wu [19] extended the previous algorithm by using a more general pruning rule called Rule k: a gateway becomes a non-gateway if all of its neighbors are also neighbors of any one of k other nodes that are connected and have higher priority values. Rules 1 and 2 are special cases of Rule k where k is restricted to 1 and 2, respectively.

Span (static): Chen, Jamieson, Balakrishman, and Morris [20] proposed the *Span* protocol to construct a set of forward nodes (called *coordinators*). A node v becomes a coordinator if it has two neighbors that are not directly connected, indirectly connected via one intermediate coordinator, or indirectly connected via two intermediate coordinators. Before a node changes its status from non-coordinator to coordinator, it waits for a backoff delay which is computed from its energy level, node degree, and the number of pairs of its neighbors that are not directly connected. The backoff delay can be viewed as a priority value, such that nodes with shorter backoff delay have a higher chance of becoming coordinators.

LENWB (dynamic): Sucec and Marsic [21] proposed the *Lightweight and Efficient Network-Wide Broadcast* (LENWB) protocol, which computes the forward node status on-the-fly. Whenever node v receives a broadcast packet from a neighbor u, it computes the set C of nodes that are connected to u via nodes that have higher priority values than v. If v's neighbor set, $N(v)$ (i.e., $N_1(v)$) is contained in C, node v is a non-forward node; otherwise, it is a forward node.

SBA (dynamic): Peng and Lu [16] proposed the Scalable Broadcast Algorithm (SBA) to reduce the number of forward nodes. As in LENWB, the status of a forward node is computed on-the-fly. When a node v receives a broadcast packet, instead of forwarding it immediately, v will wait for a backoff delay. For each neighbor u that has forwarded the broadcast packet, node v removes $N(u)$ from $N(v)$. If $N(v)$ does not become empty after the backoff delay, node v becomes a forward node; otherwise, node v is a non-forward node.

Stojmenovic's algorithm (hybrid): Stojmenovic, Seddigh, and Zuinic [17] extended Wu and Li's algorithm in two ways: (1) Suppose every node knows its accurate geographic position, only 1-hop information is needed to implement the marking process and Rules 1 and 2. That is, each node only maintains a list of its neighbors and their geographic positions (connections among neighbors can be derived). (2) The number of forward nodes are further reduced by a neighbor elimination algorithm similar to the one used in SBA.

The difference among above special cases is illustrated by Figure 13. To have a fair comparison, each node is equipped with only 2-hop information. Node u in subgraphs (a), (b), and (c) can be pruned by Wu and Li's algorithm. Node u in subgraphs (a) to (d) can be pruned by Dai and Wu's algorithm. Node u in subgraphs (a), (b), (c), and (e) can be pruned by Span. Node u in subgraphs (a) to (f) can be pruned by LENWB. Node u in subgraphs (a) and (g) can be pruned by SBA. Node u in subgraphs (a), (b), (c), and (g) can be pruned by Stojmenovic's algorithm. Node u in all subgraphs can be pruned by the coverage condition. A static protocol has only gray nodes, whereas a dynamic protocol has both gray and black nodes.