

Efficient Broadcasting using Packet History in Mobile Ad-Hoc Networks

Spiros Agathos¹ and Evangelos Papapetrou²

Department of Computer Science, University of Ioannina, Dourouti Campus, 45110, Greece

email: sagathos@cs.uoi.gr¹, epap@cs.uoi.gr²

Abstract

Broadcasting is a key component of several networking mechanisms in mobile ad hoc networks. However, its traditional implementation suffers from resource consumption caused by message redundancy. Several efficient algorithms have been proposed to tackle this problem. Among them, the class of algorithms that construct on-demand a connected dominating set, known as dominant pruning, yields an improved performance. Dominant pruning algorithms capitalize on neighborhood information and the previous hop of the broadcast message in order to minimize the set of nodes that forward a message. In this paper, the proposed algorithm makes the observation that all the nodes, visited by a broadcast message, may be exploited in order to minimize the cost of broadcasting. The analysis of the proposed algorithm as well as extensive simulation results confirm the rationale behind our approach and prove the performance gains.

1 Introduction

Nowadays the use of wireless mobile ad-hoc networks (MANETs) for supporting temporary communication needs has been spread in various application scenarios. However, protocols used in wired networks cannot be ported in wireless networks due to mobility and the nature of wireless transmission. In wired networks, topology remains constant while in wireless mobile networks nodes can move to any direction producing topology variations. As a result, communication is hard or even impossible to be established. Many wireless networking protocols use broadcasting for adapting to topology variations. For example, many routing [1] and multicast [2] algorithms use network layer broadcasting in order to discover a new route or send a route invalidation message. Therefore, a cost-effective broadcasting mechanism is required. Flooding is the most popular technique for reliable broadcasting due to its simple implementation. However, a study of pure flooding brought to surface the *broadcast storm problem* [3] and made clear that the limited bandwidth and energy resources necessitate the design of efficient communication protocols.

Several research groups have dealt with efficient broadcasting. The proposed algorithms can be classified into three categories: a) *heuristic*, b) *cluster based*, and c) *minimum connected dominating set approximations*. The objective of all schemes is to eliminate redundant transmissions by minimizing the set of nodes that forward packets. Heuristic schemes [3–9] utilize probabilistic parameters that depend mainly on topology-based criteria in order to determine the nodes allowed to forward packets. The drawback of heuristic algorithms is the difficulty in fine-tuning the probabilistic parameters [10] in order to guarantee message delivery even to poorly connected networks. In cluster based schemes [11], broadcasting is performed by cluster heads and gateway nodes. More specifically, cluster heads are responsible for delivering packets inside clusters, whereas gateways propagate packets between clusters. However, in this approach, cluster heads and gateways become points of congestion. As a result, cluster-based broadcasting may not fit with some applications; For example, when flooding is used as part of an on-demand routing protocol, congestion points should be avoided. Furthermore, clusters are susceptible to mobility therefore their maintenance induces increased cost.

Another approach is to use a *connected dominating set (CDS)* for broadcasting [12–25]. The idea is to permit only nodes that belong to the CDS to forward packets. The objective is to reduce the size of the calculated CDS in order to achieve optimal transmission overhead. The CDS may be constructed on a proactive [12–14, 17, 18] or reactive (on-demand) fashion [19–23] or may be based on GPS information [24, 25]. Hybrid approaches that combine heuristic and CDS-based approaches have also been proposed [26]. Recent works include CDS-based approaches that incorporate network-coding [27] and directional antennas [28]. In this paper, we focus on the subclass of algorithms that construct a CDS on an on-demand basis, known as dominant pruning. This class of algorithms has been identified to provide a flexible and low overhead solution [29]. In dominant pruning, each node v , receiving a broadcast message from node u , determines a CDS valid for its 2-hop neighborhood by electing 1-hop neighbors that are allowed to forward the message. The election is carried out by a pruning process that exploits the neighborhood of nodes v and u in order to minimize the calculated CDS [20], [21]. The approach is based on the observation that the neighbors of u should not be considered because they have already received the message.

Although the dominant pruning approach has been proved to be effective, using only the neighborhood of the previous hop node clearly provides an underestimation of the nodes that have already received the message. Providing a better estimation should result in performance improvement. In this paper, we make the observation that the set of nodes visited by a message in its travel from the source node to a node v , provides a good basis for estimating the nodes that have already received this message. Therefore, the set of these nodes may be utilized by the pruning process in order to minimize the constructed CDS. The proposed algorithm, called *History-based 2-hop Dominant Pruning (H2DP)*, exploits the neighborhood of each node visited by a broadcast message in order to ameliorate broadcasting. We provide an analysis of H2DP that justifies our approach while an extensive simulation study confirms its improved performance compared to TDP, PDP [21] and DP [20] protocols, which are

the most representative of the dominant pruning category and yield the best performance. Finally, we propose an optimization of H2DP, called HDP, which is tailored for sparse and highly mobile networks.

The rest of this paper is organized as follows. In Section 2, we provide useful notation and briefly describe dominant pruning. Then, in Section 3, H2DP is delineated while its complexity analysis is presented and discussed. In Section 4, we describe the model used in the simulation study while in Section 5 simulation results are presented and discussed. The optimization of H2DP for sparse and highly mobile networks is presented in Section 6. Finally, in Section 7 useful conclusions are drawn.

2 Preliminaries

H2DP falls into the category of dominant pruning algorithms. Therefore, before delineating its operation, we formulate the problem of dominant pruning. To this end, the following notation is employed:

- $N(u)$ denotes the set of nodes that lie 1-hop away from node u , including u itself, and is called the 1-hop neighborhood of u .
- $N(N(u))$ denotes the 2-hop neighborhood of node u , i.e., the set of nodes that lie within 2 hops from node u . Clearly, $N(u) \subseteq N(N(u))$.

In dominant pruning, each node v maintains knowledge about $N(N(v))$ by exchanging with its neighbors periodic hello messages that contain 1-hop neighborhood information. In order to forward a broadcast message, node v calculates a subset of $N(v)$, called the *forwarding set*, and piggybacks it on the message.

Definition 1. Forwarding set of node v ($Fw(v)$): *The subset of $N(v)$ consisting of neighbors that are allowed to forward a message.*

In dominant pruning’s approach, $Fw(v)$ is a CDS of the graph $N(N(v))$, therefore all nodes in $N(N(v))$ will receive the message after it is forwarded by all nodes in $Fw(v)$. Each node $w \in Fw(v)$, after receiving the message, repeats the procedure. The CDS for the entire network consists of the forwarding sets computed locally by each node. The algorithm terminates either when $w \notin Fw(v)$ or when node $w \in Fw(v)$ but a *termination criterion* is met. Two termination criteria have been proposed [21], namely *relayed/unrelayed* and *marked/unmarked*. The latter specifies that node w should not forward the message if all nodes in $N(w)$ have already received it, while the former mandates that node w forwards a message only once. In this paper we will focus on the *relayed/unrelayed* criterion since it provides a low cost implementation compared to the *marked/unmarked* criterion [21].

The calculation of $Fw(v)$ at node v , for a packet coming from node u , involves two sets, namely $B(v)$ and $U(v)$.

Definition 2. Broadcast candidate set of node v ($B(v)$): *A subset of $N(v)$ consisting of nodes that are candidates for forwarding a message.*

Definition 3. Universal¹ set of node v ($U(v)$): A subset of $N(N(v))$ comprising nodes that should receive the broadcast message after it is forwarded by all nodes in $Fw(v)$.

The set $B(v)$ can be seen as a set of sets if each node $w \in B(v)$ is replaced by $N(w) \cap U(v)$. In this respect, the calculation of $Fw(v)$ can be formulated as a set cover problem, the minimization version of which is known to be NP-hard [30]. The *greedy set cover (GSC)* algorithm can be used to produce $Fw(v)$. GSC is a well-known polynomial time algorithm [30], [31] that adopts a greedy approach in order to provide an approximation of the optimal solution. Although all dominant pruning algorithms adopt GSC, they take a different approach in calculating $B(v)$ and $U(v)$. This is because the performance of GSC in terms of $Fw(v)$ minimization and time complexity depends on the size of $B(v)$ and $U(v)$. The most representative algorithms are:

- *Dominant Pruning (DP)* [20], in which $U(v) = N(N(v)) - N(u)$,
- *Partial Dominant Pruning (PDP)* [21], where $U(v) = N(N(v)) - N(N(v) \cap N(u))$,
- *Total Dominant Pruning (TDP)* [21] with $U(v) = N(N(v)) - N(N(u))$

At the same time, $B(v) = N(v) - N(u)$, for all schemes. TDP defines the smallest $U(v)$ set therefore minimizes the number of elected forward nodes while having the smallest time requirements. In Section 3.1, we will provide more insight on the performance of dominant pruning schemes.

3 History-based 2-hop Dominant Pruning (H2DP)

H2DP falls into the category of algorithms that construct on-demand an approximation of a minimum CDS by utilizing the dominant pruning approach. Moreover, H2DP introduces the concept of *packet history*. Let us consider the case that a packet² p is received by a node v after being transmitted by node u . The packet history consists of the nodes that have already received p . More specifically, the history $H^v(p)$ is defined as:

Definition 4. Packet History $H^v(p)$: The set of nodes that have received packet p in its travel along $P^v(p)$.

while the *packet route* $P^v(p)$ is defined as:

Definition 5. Packet route $P^v(p)$: The set of forward nodes that packet p has visited from the source node s to node v , including v .

¹The term universal refers to the vicinity of v and not to the entire network

²The term packet is used to refer to a copy of a broadcast message. However, in some parts the terms packet and message are used interchangeably.

According to H2DP's approach, reducing the size of the universal set $U(v)$ holds a central role in minimizing the number of elected forwarding nodes and therefore in alleviating the cost of broadcasting. To this end, H2DP makes the observation that $U(v)$ may contain nodes that have already received p (i.e., they belong to the packet's history), therefore such nodes may be eliminated from $U(v)$. More specifically, upon reception of p coming from node u , v examines the forwarding list $Fw(u)$ which is piggybacked on p 's header. If $v \notin Fw(u)$ then the packet is discarded. Otherwise, v calculates $B(v) = N(v) - N(u)$ and:

$$U(v) = N(N(v)) - H^u(p) \quad (1)$$

and constructs $Fw(v)$ by utilizing GSC. In the special case that v is the source node s , $B(s) = N(s)$ and $U(s) = N(N(s))$. Determining the nodes in packet history is clearly very important for the performance of H2DP. The simplest approach is to consider only the nodes in $P^u(p)$, i.e., $H^u(p) = P^u(p)$. This approach has been proposed in the literature of non dominant pruning schemes [32] on the basis that only those nodes are proved to have received p . However, this approach provides an underestimation of the nodes that have actually received the packet and has a negligible impact on reducing $U(v)$ since usually $|P^u(p)| \ll |U(v)|$. On the other hand, H2DP makes the observation that each node w in $P^u(p)$ is a forwarding node. Therefore, all nodes in $N(N(w))$ have been scheduled to receive the packet by the execution of the algorithm at w . Consequently, in H2DP:

$$H^u(p) = \bigcup_{\forall w \in P^u(p)} N(N(w)) \quad (2)$$

After calculating $Fw(v)$, node v updates packet history:

$$H^v(p) = H^u(p) \cup N(N(v)) \quad (3)$$

and piggybacks $Fw(v)$ and $H^v(p)$ on the packet's header. The algorithm terminates either when p has been forwarded by v in the past or when $v \notin Fw(u)$. The concept of history in H2DP and its impact on the determination of $U(v)$ is depicted in Fig. 1(a). In H2DP, node v makes the assumption that, given a node $w \in P^u(p)$, all nodes in $N(N(w))$ have received packet p . This is the same assumption as the one made by TDP if w is the previous hop node (Fig. 1(b)). However, H2DP results in a smaller universal set $U(v)$ because it makes the aforementioned assumption $\forall w \in P^u(p)$. The correctness of H2DP's approach can be proved by the following theorem:

Theorem 1. *If a node $w \in N(N(v))$ and $w \in H^u(p)$ then w can be excluded from $U(v)$.*

Proof. Since node $w \in H^u(p)$ this means that there exists a node $f \in P^u(p)$ for which $w \in N(f)$ or $w \in N(N(f))$. In the first case the transmission by f , which is a forward node, delivers the packet to w . In the second case, f has elected a forwarder for covering w therefore w receives the packet after the transmission by this forwarder. In the special case that f is the previous hop node ($f = u$) and node v is elected as the forwarder for covering w this means that $w \in N(v)$, therefore the transmission by v delivers the message to w . If the termination criterion at v has already been met (v has already forwarded p) this means that all nodes in $N(v)$ (including w) have already received the packet. \square

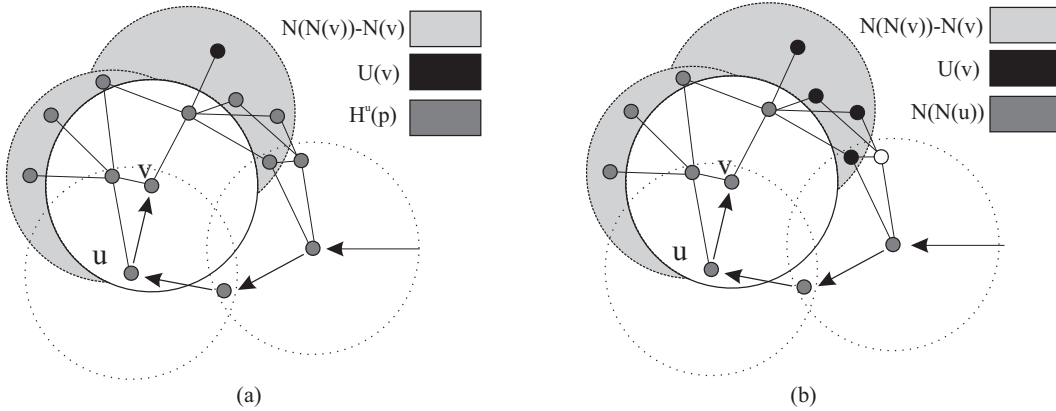


Figure 1: Illustration of: a) H2DP, and b) TDP when executed at a node v . Arrows indicate the path followed by the broadcast message.

In the following, it will be shown that reducing $U(v)$ allows H2DP to minimize the number of nodes that need to forward a message while at the same time the algorithm is executed in a smaller amount of time. Both of these characteristics are very important in the energy constrained context of a MANET. Moreover, reducing the number of transmissions economizes on scarce network resources, such as bandwidth, and alleviates contention and/or congestion on the wireless medium.

3.1 Complexity analysis

All dominant pruning algorithms, including H2DP, rely on GSC to implement the forwarder selection process. GSC provides an approximation of the optimal solution since the set cover problem is known to be NP-hard [30]. Consequently, all algorithms inherit the approximation nature of GSC. However, the performance and time complexity of each dominant pruning scheme depends on the sets B and U that each algorithm provides as input to GSC. In the following, we analyze the performance of H2DP when executed at a node v and compare it to other dominant pruning schemes.

Time complexity: GSC is a polynomial time algorithm known to run in time [30]:

$$O(|U(v)||B(v)| \min\{|U(v)|, |B(v)|\}) \quad (4)$$

However, a linear time implementation is possible [30], [31] that runs in time:

$$O\left(\sum_{\forall v_i \in B(v)} |N(v_i) \cap U(v)|\right) \quad (5)$$

Theorem 2. *H2DP has an improved time complexity compared to TDP, PDP and DP.*

Proof. Let us define by $U^H(v)$ and $U^T(v)$ ³ the universal sets for H2DP and TDP, respectively. It is clear from eq. 1 that $U^H(v) \subseteq U^T(v)$ because $N(N(u)) \subseteq H^u(p)$. Since $B(v)$ is common for both

³In the following we will use the superscripts H, T, P and D to indicate that the referenced quantity refers to H2DP, TDP, PDP and DP, respectively

schemes:

$$(N(v_i) \cap U^H(v)) \subseteq (N(v_i) \cap U^T(v)), \forall v_i \in B(v) \quad (6)$$

Therefore, $|(N(v_i) \cap U^H(v))| \leq |(N(v_i) \cap U^T(v))|$, $\forall v_i \in B(v)$ and based on eq.5 it is clear that the time complexity of H2DP is better than that of TDP. Similarly, it can be proved that H2DP has an improved time complexity compared to PDP and DP. \square

Forwarding complexity: The cost of broadcasting is related to the number of elected forwarding nodes. GSC is used for the election process, providing an approximation of the optimal solution with an approximation factor ρ known to be [30]:

$$\rho = H(\max\{|N(v_i) \cap U(v)| : \forall v_i \in B(v)\}) \quad (7)$$

where $H(k) = \sum_{i=1}^k \frac{1}{i}$. Since $\sum_{i=1}^k \frac{1}{i} \leq \ln(k+1)$ and $|N(v_i) \cap U(v)| \leq |U(v)|, \forall v_i \in B(v)$, the approximation factor ρ can also be expressed as [30]:

$$\rho = \ln(|U(v)| + 1) \quad (8)$$

Eqs. 7 and 8 prove that the performance of each dominant pruning scheme at node v depends on the determination of $B(v)$ and $U(v)$.

Theorem 3. *The approximation factor ρ^H of H2DP is better or equal to the approximation factor ρ^T of TDP.*

Proof. In the case that node v is the source node s or $v \in N(s)$, $U(s)$ is the same for both schemes. More specifically, $U^H(s) = U^T(s) = N(N(s))$, while $U^H(v) = U^T(v) = N(N(v)) - N(N(s)), \forall v \neq s \in N(s)$. Consequently, according to eq. 7, the approximation factor for both schemes is the same. On the contrary, when the dominant pruning algorithm is executed at any node $v \notin N(s)$ for a packet p received from node u then:

$$\begin{aligned} U^H(v) &= N(N(v)) - H^u(p) \\ &\subseteq N(N(v)) - N(N(u)) = U^T(v) \end{aligned} \quad (9)$$

Consequently, $|U^H(v)| \leq |U^T(v)|$ and according to eq. 7 $\rho^H \leq \rho^T$. \square

Similarly, it can also be proved that H2DP exhibits a better approximation factor than that of PDP and DP.

3.2 Representation of Packet History and Neighborhood Information

The method used for representing the packet history is a key performance factor of H2DP. The reason is twofold; On one hand, the history size affects the packet size. The simulation study presented in the following, confirms that even if a raw representation method is used (i.e., each node is represented by its address), H2DP produces packets of reasonable size even for large networks. However, reducing the

packet size is a challenging objective which should be combined with the alleviation of computational overheads related to operations on the packet history and/or the universal set such as search, addition and deletion. Eqs. 1-3 indicate that a node is frequently involved in such operations. For example, suppose that a node v receives a packet p from node u . The calculation of $U(v)$ involves searching whether each node $w \in N(N(v))$ is a member of $H^u(p)$. Such computational overheads also exist in all other dominant pruning schemes but have never been addressed.

It is clear that a more elaborate representation method for packet history and possibly for neighborhood information should provide size minimization as well as fast search, addition and deletion operations. To this end, we propose the use of bit arrays or bit vectors. Bit arrays are the simplest way to implement direct addressing [30] in which dictionary operations are executed in $O(1)$ time. In our implementation, a bit vector of size S may be used to represent a set of S nodes. A bit set to 1 indicates that a node is a member of the set. The network addressing scheme, e.g., IP addressing, may be used to provide an one-to-one mapping of node addresses to bit vector indexes. This mapping can be predefined if all possible members of the set are known. It is clear that all network nodes may be members of a packet history. Therefore, in order to use bit vectors the size of the vector S should be equal to the maximum network size, i.e., $S = N$. This results in a requirement of N bits for a network of N nodes. For a 200 node network the required history size is 25 bytes which is rather small; only a set of 6 nodes could be represented with the same amount of bytes if IP addressing is assumed and the whole address is stored. Therefore, bit arrays may also provide efficient compression of packet history.

4 Simulation Environment

Set up and methodology: The performance of H2DP is assessed through comparison with TDP, PDP and DP algorithms which are known to yield the best performance in the dominant pruning class of algorithms. In broadcast algorithms, like those investigated in this paper, simulating realistic network conditions is of great importance. This is because packets are transmitted through MAC layer broadcasts, therefore packet collisions result in packet drops. Collisions also have an impact on hello messages and therefore affect the accuracy of the neighborhood information. Furthermore, congestion may also result in packet losses and/or delay. In this work, we used the medium access control of IEEE 802.11, provided in the CMU extension of ns2 [33], in order to approximate realistic network conditions. Moreover, we used configurations of high mobility in order to assess the resilience of the algorithms to stale neighborhood information. All presented results were obtained as average values over 20 independent simulation runs. The simulation time for each run was 900 seconds. The confidence level for all reported results is 95%. The simulation parameters are summarized in Table 1.

Network model: The network consists of N nodes that roam in a square area. The transmission range of each node is set to $R = 250m$ while the nominal bitrate is set to $2Mb/s$. Mobility scenarios have been produced using the well-known Random Waypoint (RW) Algorithm and the perfect simulation

Table 1: Simulation parameters

Simulation Time	900 secs
Number of Trials	20
Confidence level	95%
Number of nodes (N)	50 - 200
Maximum speed	1-20 m/secs
Pause time in RW	0 secs
Transmission Range (R)	250 m
Bandwidth	2 Mb/s
Broadcast sessions	30
Broadcast Rate (λ)	0.1 - 2 pkts/s
Message size	256 bytes
Hello interval (t_H)	0.5 - 5 secs

model [34] in order to avoid transient artifacts in nodes' movement. We consider a "sparse" and a "dense" network topology. The "sparse" topology corresponds to a low density network in which a node has only one neighbor in the worst case while the average number of neighbors is 15. On the other hand, in the "dense" network topology the node density is twice the density of the "sparse" topology, i.e. the average number of neighbors is 30. In order to control the node density and produce "sparse" and "dense" topologies, we used square roaming areas of different size depending on the number of nodes N . The actual size of the roaming area is derived through the node distribution proposed for the Random Waypoint algorithm in [35] and the calculation of node degree proposed in [36].

Packet generation: Thirty broadcast sessions are considered in total, each one stemming from a different source node. Each session periodically produces broadcast messages at a rate λ . The size of each broadcast message has been set to 256 bytes.

Algorithm specific parameters: Each node maintains 2-hop neighborhood information by periodically exchanging hello messages with an interval of t_H seconds. A neighbor is removed if a hello message has not been received within $2t_H$ seconds since the last update. An important issue related to broadcast schemes is the jittering required for avoiding collisions of duplicate packets [29]. In order to have a fair comparison of all algorithms we chose a random jittering scheme with the maximum jitter set to 0.1 secs. Regarding history representation, the optimization presented in Section 3.2 has not been used in order to capture the worst-case performance of H2DP. Instead, we used 4 bytes for representing a node in the forwarding set or in the packet history. This size is sufficient for storing addresses even if an IP-based addressing scheme is considered.

5 Simulation Study and Analysis

In the first experiment, we evaluate the performance of H2DP versus the network size in dense and sparse networks under low mobility (1 m/s) and traffic load (0.2 $pkts/s$). Fig. 2 illustrates the *number of forward nodes*, i.e., the average number of nodes that forward a broadcast message, for all the investigated algorithms. H2DP outperforms all other schemes for all network sizes in both the dense and the sparse topologies. H2DP’s performance in small networks (less than 75 nodes) is almost equivalent to that of TDP (and PDP), especially in the case of dense networks. The reason is that most nodes are within two hops from the source node s . As a result, H2DP works mainly in the first hop, therefore the history of a packet p at a node $v \in N(s)$ consists only of $N(N(s))$, i.e., $H^s(p) \equiv N(N(s))$. In this case, according to eq. 1, the operation of H2DP is identical to that of TDP and close to the operation of PDP. When the network size increases, many nodes are located more than one hop away from s , i.e., a route $P^v(p)$ to a node v contains more hops on average. Therefore, according to eq. 2, the packet history is populated by more nodes. Consequently, $U(v)$ is effectively reduced and the approximation factor of GSC improves (see eq. 7), resulting in reduced forwarding sets. On the other hand, the improvement of TDP and PDP compared to DP is smaller than that of H2DP and remains approximately the same for all network sizes. This can be ascribed to the strategy of both algorithms to only take advantage of the previous hop u in order to minimize the universal set at node v (TDP only uses $N(N(u))$ while PDP utilizes $N(N(v) \cap N(u))$). This strategy cannot take advantage of the longer paths that a packet travels as the network size increases. TDP reduces forward nodes by $\sim 22\%$ compared to DP, PDP improves by $\sim 14\%$ while H2DP manages a gain up to $\sim 46\%$. The improvement compared to DP is less intense for all schemes in the sparse topology ($\sim 26\%$, $\sim 14\%$ and $\sim 10\%$ for H2DP, TDP and PDP respectively) due to the reduced neighborhood size.

It must be noted that the tradeoff for the improvement witnessed in TDP, PDP and H2DP with respect to DP is the increased computational and space requirements due to the extensive use of 2-hop neighborhood information. However, the representation method proposed in Section 3.2 may be used in order to effectively alleviate the space as well as the computational requirements related to the use of 2-hop neighborhood.

Fig. 3 depicts the *delivery ratio*, i.e., the percentage of network nodes that receive a broadcast message. This metric is of great importance in contention based and mobile networks. All schemes present a high delivery ratio near to 100% for small dense networks (Fig. 3(a)). However, in large dense networks there is a rapid performance degradation due to congestion and packet collisions. Since the packet generation rate is relatively low (only 0.2 $pkts/s$), congestion is the result of packet redundancy produced by the algorithms. This is confirmed by Fig. 4 where the *packets received per node*, i.e., the average number of duplicate packets of a broadcast message received by a node, is illustrated. Indeed, more duplicate packets are produced for large dense networks since there are more forwarding nodes (Fig. 2(a)) and more disjoint paths through which packet duplicates may reach a

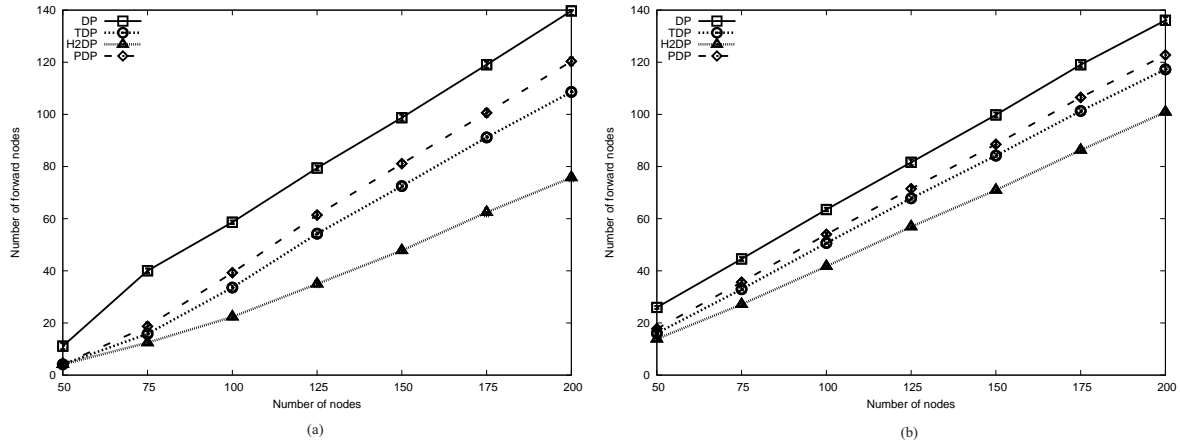


Figure 2: Number of forward nodes versus network size (1 m/s , 0.2 pkts/s , $t_H = 2\text{ secs}$) for: a) dense topology, and b) sparse topology

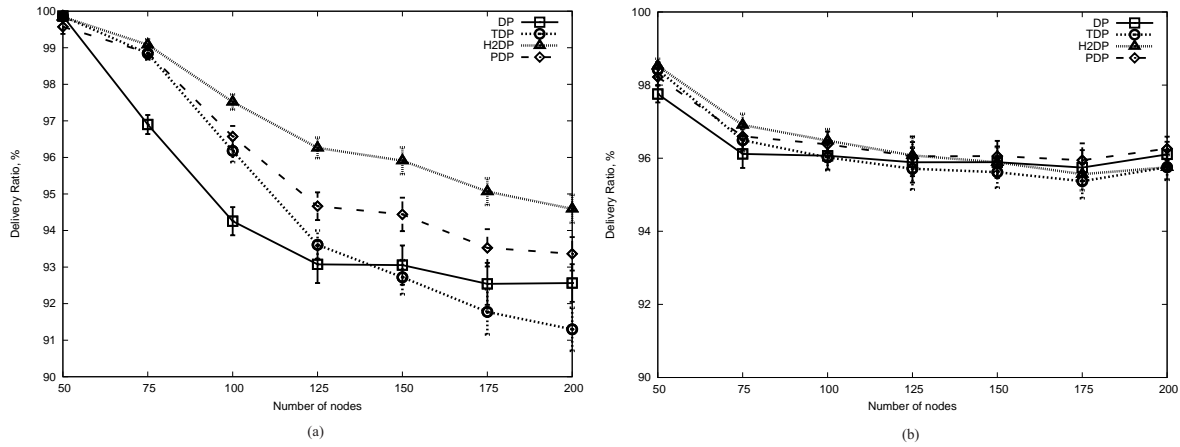


Figure 3: Delivery ratio versus network size (1 m/s , 0.2 pkts/s , $t_H = 2\text{ secs}$) for: a) dense topology, and b) sparse topology

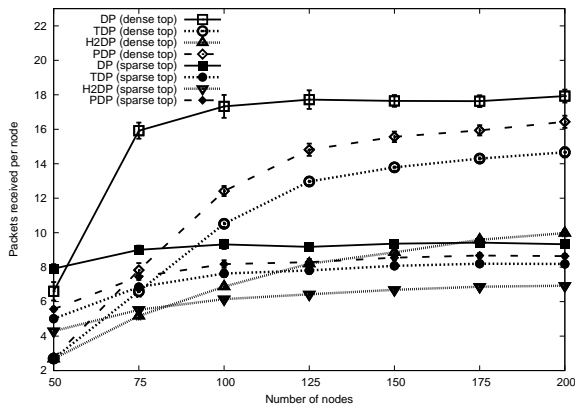


Figure 4: Number of packets per node versus network size for both dense and sparse topologies

node. In small networks, both H2DP and TDP achieve better delivery ratio compared to DP since both protocols minimize congestion. On the contrary, in large networks TDP fails to reduce congestion to the same extent as H2DP does (Fig. 4). The same applies to DP and PDP, however their increased packet redundancy compensates for packet losses due to congestion and results in a slower degradation of the delivery ratio compared to TDP. However, this is not enough to outperform H2DP which effectively reduces network load even for large networks and achieves the highest delivery ratio. Similar performance characteristics are witnessed for sparse networks. However, differences are smoothed out since the reduced connectivity induces less congestion for all schemes. Finally, concerning Fig. 4, we should point out that packet redundancy is related to the node’s energy consumption since packet reception and processing involves an energy cost. Consequently, Fig. 4 reveals another advantage of H2DP.

Fig. 5(a) illustrates *mean delay*, i.e., the average delay involved in receiving a broadcast message, which is another important performance metric. In dense networks, H2DP induces smaller delays by minimizing the number of duplicate packets. This minimization has a twofold impact; on one hand alleviates congestion and therefore smaller queueing delays occur while on the other hand medium access delay is minimized as a result of less contention. The increasing trend of delay with respect to network size is ascribed to the increasing number of hops involved in a broadcast. As a result, longer delays are evident in sparse networks where paths consist of more hops due to the reduced connectivity. In this case, the performance of all schemes is similar since delay is dominated by the number of hops.

Finally, in Fig. 5(b) we present the *normalized byte overhead*, i.e., the average number of bytes transmitted in the network during the broadcast of a message. This metric evaluates the impact of piggybacking the history on the packet’s header. Surprisingly, H2DP induces less bytes in the dense topology scenario while its performance is close to the other schemes in sparse networks. Bear in mind that the presented results represent the worst case scenario for H2DP since the optimization for

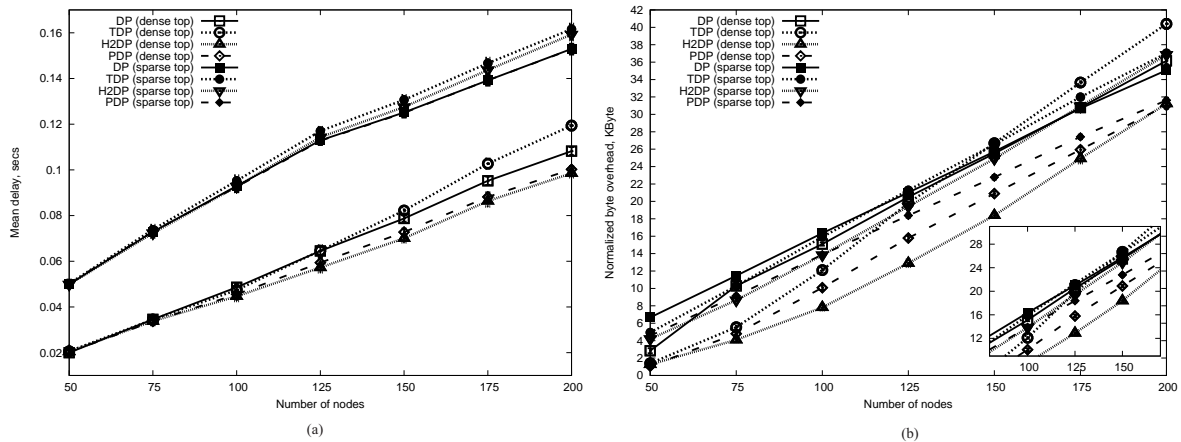


Figure 5: Performance versus network size (1 m/s , 0.2 pkts/s , $t_H = 2\text{ secs}$): a) Mean delay, and b) Normalized byte overhead

packet history representation, proposed in Section 3.2, has not been used. Instead, we used 4 bytes for each address piggybacked on the packet’s header. The significantly reduced number of transmissions (Fig. 2) justifies H2DP’s performance. Although packets of more bytes are created, the overall balance is positive. It should be pointed out that, in contention-based networks, the bandwidth is consumed by the access delay which is dominant compared to the transmission delay. Keeping in mind that access delay depends on the number of packet transmissions while transmission delay depends on the packet size, it is clear that minimizing the number of transmissions has a greater impact on bandwidth savings.

5.1 Resilience to mobility

It is important for schemes that use neighborhood information, such as dominant pruning, to evaluate their resilience to high mobility. Therefore, in this section we will investigate the impact of: a) node mobility and b) hello interval, i.e. the rate at which topology information is refreshed.

Impact of mobility: In Fig. 6(a) the delivery ratio for all algorithms is presented versus node mobility for sparse and dense networks. In sparse networks, mobility severely affects all algorithms since neighborhood information is quickly becoming stale. The impact of mobility is greater in TDP and H2DP. More specifically, although H2DP is superior to DP (by $\sim 0.4\%$) when mobility is low, its performance lacks $\sim 0.4\%$ when mobility is 20 m/s . Similar performance is witnessed for TDP. This behavior is ascribed to the fact that 2-hop neighborhood information, which has an increased invalidation rate compared to 1-hop neighborhood, is used to minimize $U(v)$. However, in dense networks, H2DP preserves its advantage over all algorithms through the entire mobility range. In this case, although neighborhood information is invalidated, the increased node connectivity provides

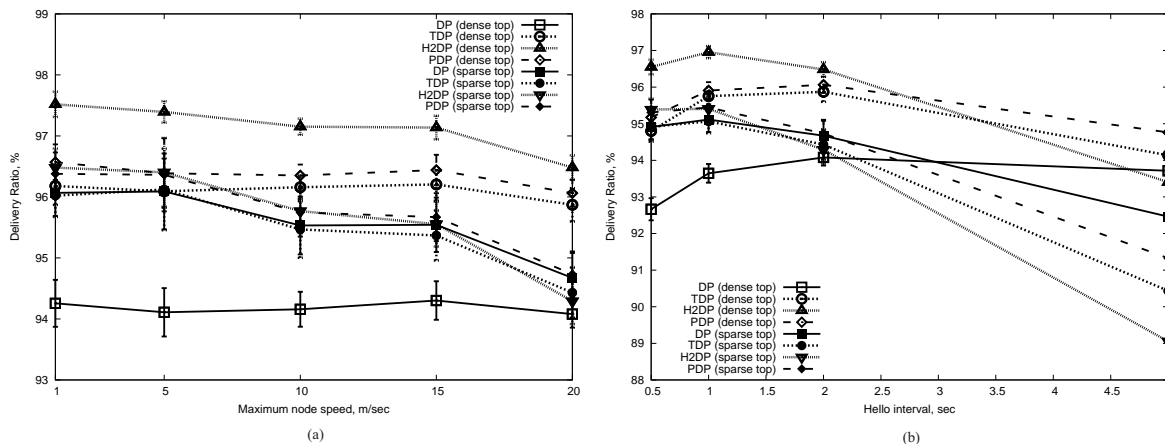


Figure 6: Delivery ratio vs: a) mobility (100 nodes, 0.2 pkts/s, $t_H = 2$ secs), and b) hello interval (20 m/s, 100 nodes, 0.2 pkts/s) for dense and sparse networks

multiple paths (path diversity) that enable all algorithms to efficiently deliver the broadcast message. These results indicate that algorithms utilizing 2-hop neighborhood may become less efficient in sparse networks of increased mobility (> 20 m/s) such as VANETs. In Section 6, we propose an optimized version of H2DP that presents increased resilience to mobility.

Impact of hello interval: The impact of mobility depends on the refresh rate of neighborhood information, i.e., the hello interval. A short hello interval provides the means for dealing with performance degradation in highly mobile networks. In Fig. 6(b), the delivery ratio is presented versus the hello interval for all algorithms in networks of high mobility (20 m/s). As expected, long hello intervals cause degradation in performance of all schemes; however the impact is more intense on H2DP and TDP because 2-hop neighborhood is used. Again, the impact of hello interval is heavier in sparse networks due to the reduced path diversity. Shorter hello intervals are in favor of H2DP in both dense and sparse networks. Although all schemes improve their performance, due to more accurate neighborhood information, the improvement for H2DP is sharper since it is combined with reduced levels of congestion due to the minimization of elected forwarding nodes. On the other hand, all other schemes produce more transmissions therefore part of their improvement is nullified by increased levels of collisions. This is evident in a more clear way in the case of dense networks where congestion is heavier than in sparse networks. In this case, only H2DP improves its performance for hello intervals of 1-2 secs. On the contrary, all other schemes do not improve due to congestion. Finally, when the hello interval is 0.5 sec (more hello messages therefore more collisions) the performance of all schemes is degraded. It is worth mentioning that the congestion in DP is so high that shorter hello intervals have the opposite effect, i.e., the reduction of delivery ratio. According to the presented results it is clear that H2DP may utilize shorter hello intervals in sparse and highly mobile environments in order to improve its performance in terms of delivery ratio. The cost of using a shorter hello interval may

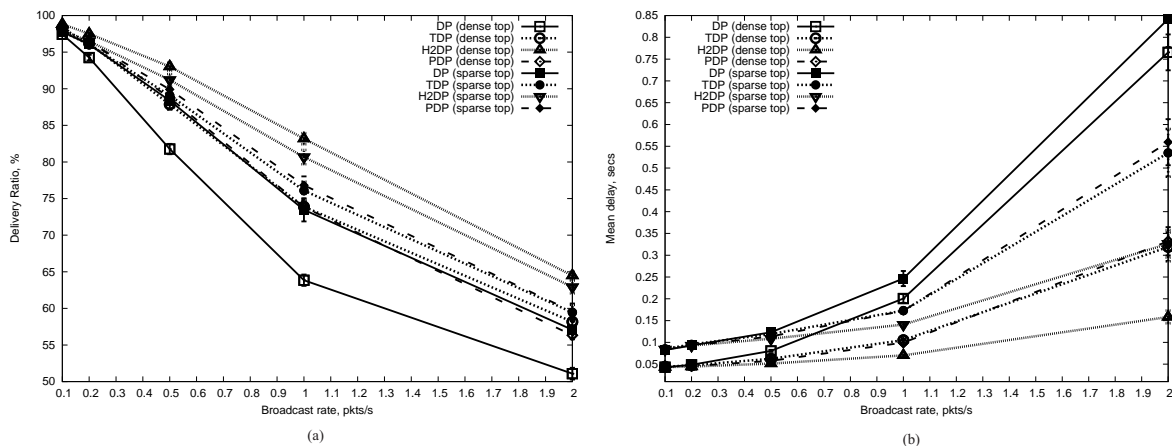


Figure 7: Performance versus broadcast rate (1 m/s , 100 nodes, $t_H = 2secs$): a) Delivery ratio and b) Mean delay

be compensated by the reduced packet redundancy witnessed in Figs. 2 and 4.

5.2 Performance under congestion

In this section we evaluate the performance under conditions of increased congestion. Such conditions are frequently present in MANETs due to the limited bandwidth. In Fig. 7, the performance of all algorithms is presented versus the broadcast rate. As expected, all schemes are heavily affected. As far as the delivery ratio is concerned (Fig. 7(a)), DP rapidly collapses, especially in dense networks since the increased network connectivity aggravates congestion. H2DP produces less duplicate packets and therefore is able to cope with congestion more effectively. As a result, H2DP achieves an improvement of $\sim 3.5\%$ in the sparse and $\sim 6.5\%$ in the dense network compared to TDP, which is the second performer. The improvement compared to DP reaches up to $\sim 13.5\%$ in the dense network and $\sim 6\%$ in the sparse. A confirmation of congestion is the mean delay, presented in Fig. 7(b). When broadcast rate is above 1 pkt/s , delay rapidly increases because longer queues are formed and frequent activity in the medium increases access delay. When the network is heavily congested (2 pkt/s) only H2DP achieves an acceptable delay (~ 160 msecs) and at the same time exhibits the best delivery ratio (Fig. 7(a)).

6 History based Dominant pruning (HDP)

The performance evaluation of H2DP has brought to light its reduced resilience to highly mobile ($> 20 m/s$) and sparse networks. This behavior stems from the fact that H2DP, when building $H^v(p)$ in node v , makes the optimistic assumption that every node in $N(N(v))$ has already or will receive p . However, stale neighborhood information, which is frequent in highly mobile environments, may

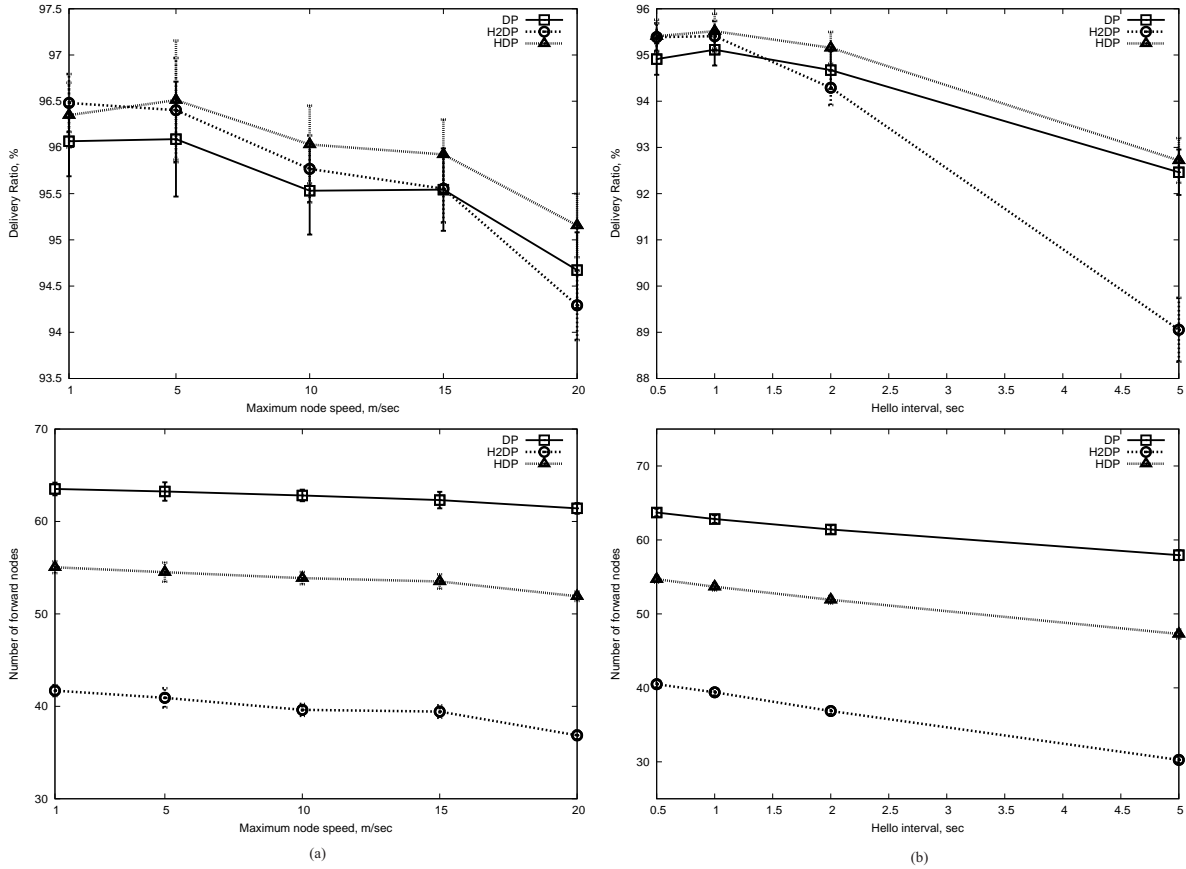


Figure 8: Delivery Ratio versus a) maximum node speed ($t_H = 2secs$), and b) hello interval ($20 m/s$) for the sparse topology scenario ($100 nodes$)

invalidate the aforementioned assumption. To overcome this shortcoming we propose to use only $N(v)$ in forming the packet history, i.e.:

$$H^v(p) = \bigcup_{\forall w \in P^v(p)} N(w) \quad (10)$$

The intuition is that resilience to mobility will be increased since $N(v)$ has a smaller invalidation rate compared to $N(N(v))$. We call the resulting dominant pruning scheme, *history-based dominant pruning (HDP)*. HDP takes a more conservative approach compared to H2DP. It only makes the assumption that all nodes in $N(w)$, $\forall w \in H^v(p)$ have already received p . Clearly, the approach of HDP is to tradeoff reduction of forward nodes for delivery efficiency. This can also be verified by eqs. 2, 10 and 7; the history of HDP is smaller than that of H2DP therefore $U(v)$ is bigger in HDP. As a result, the approximation factor is worse than that of H2DP.

Fig. 8(a) illustrates the delivery ratio and the number of forward nodes for HDP with respect to node mobility while Fig. 8(b) presents the same performance metrics versus the hello interval when mobility is fixed to $20 m/sec$. Both experiments are carried out in sparse networks. HDP is compared

to H2DP and DP which presents the best resilience to highly mobile and sparse networks. The approach of HDP is proved to be effective since it outperforms DP with respect to delivery ratio even for the highest mobility. Moreover, this is accomplished in parallel to a reduction of the number of forward nodes that ranges in ~ 13.5 - 15.5% for all mobility levels. This result confirms the tradeoff that HDP introduces since its performance lies between that of DP and H2DP. The resilience of HDP to mobility is also confirmed by Fig. 8(b) where the impact of hello interval is illustrated. Even for long intervals HDP preserves its superiority over both DP and H2DP. At the same time, even though HDP produces more forward nodes compared to H2DP, it still manages an impressive reduction that reaches up to 18.5% compared to DP.

7 Conclusions

In this paper a new approach to dominant pruning, called History-based 2-hop Dominant Pruning (H2DP), has been proposed for minimizing the redundant transmissions during broadcasting. H2DP makes the observation that the history of a packet may be used for minimizing the set of forwarding nodes in dominant pruning-based schemes. The analysis of H2DP as well as its extensive simulation study have confirmed its improved performance compared to existing dominant pruning schemes with respect not only to the minimization of broadcasting cost but also to the ability to deliver a broadcast message. Furthermore, an optimization of H2DP, tailored for highly mobile and sparse networks, has been proposed and proved to be effective in increasing the resilience to mobility.

References

- [1] Elizabeth M. Royer and Chai-Keong Toh. A review of current routing protocols for ad hoc mobile wireless networks. *IEEE Personal Communications*, 6(2):46–55, 1999.
- [2] Carlos de Morais Cordeiro, Hrishikesh Gossain, and Dharma P. Agrawal. Multicast over wireless mobile ad hoc networks: Present and future directions. *IEEE Network*, 17(1), January/February 2003.
- [3] S.Y.Ni, Y.C. Tseng, Y.S. Chen, and J.P. Sheu. The broadcast storm problem in a mobile ad hoc network. In *MobiCom*, pages 151–162, August 1999.
- [4] M. Bani Yassein, M. Ould-Khaoua, L. Mackenzie, and S. Papanastasiou. Improving the performance of probabilistic flooding in manets. In *Proc. of International Workshop on Wireless Ad-hoc Networks*, May 2005.
- [5] Q. Zhang and D.P. Agrawal. Dynamic probabilistic broadcasting in manets. *J. Parallel Distrib. Comput.*, 65:220–233, 2005.

- [6] Y. Tseng, S. Ni, and E. Y. Shih. Adaptive approaches to relieving broadcast storms in a wireless multihop mobile ad hoc networks. In *in Proc. of ICDCS'01*, pages 481–488, 2001.
- [7] S.O. al Humoud, L.M. Mackenzie, and J. Abdulai. Neighbourhood-aware counter-based broadcast scheme for wireless ad hoc networks. In *GLOBECOM Workshops, 2008 IEEE*, pages 1 –6, 30 2008-dec. 4 2008.
- [8] W. Woon and K.L. Yeung. Self-pruning broadcasting for mobile ad hoc networks. In *Global Telecommunications Conference, 2009. GLOBECOM 2009. IEEE*, pages 1 –6, 30 2009-dec. 4 2009.
- [9] O. Tonguz, N. Wisitpongphan, F. Bai, P. Mudalige, and V. Sadekar. Broadcasting in vanet. In *2007 Mobile Networking for Vehicular Environments*, pages 7 –12, may 2007.
- [10] Stefan Pleisch, Mahesh Balakrishnan, Ken Birman, and Robbert van Renesse. Mistral: efficient flooding in mobile ad-hoc networks. In *Proceedings of the seventh ACM international symposium on Mobile ad hoc networking and computing*, pages 7–14, May 2006.
- [11] Y. Yi, M. Gerla, , and T. Kwon. Efficient flooding in ad hoc networks using on-demand (passive) cluster formation. In *2nd Annual Mediterranean Ad Hoc Networking Workshop (Med-hoc-Net 2003)*, 2003.
- [12] C. Adjih, P. Jacquet, and L. Viennot. Computing connected dominated sets with multipoint relays. Tech. Rep. 4597, October 2002.
- [13] 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*, 13(1):14–25, 2002.
- [14] J. Wu and H. Li. On calculating connected dominating set for efficient routing in ad hoc wireless networks. In *Proc. of the Third International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pages 7–14, August 1999.
- [15] Jie Wu and Wei Lou. Extended multipoint relays to determine connected dominating sets in manets. pages 621 – 630, oct. 2004.
- [16] Marco Aurelio Spohn and J.J. Garcia-Luna-Aceves. Improving route discovery in on-demand routing protocols using two-hop connected dominating sets. *Ad Hoc Networks*, 4(4):509 – 531, 2006.
- [17] K.A.M. Almahorg, S. Naik, and Xuemin Shen. Efficient localized protocols to compute connected dominating sets for ad hoc networks. In *GLOBECOM 2010, 2010 IEEE Global Telecommunications Conference*, pages 1 –5, dec. 2010.

- [18] J. Wu, Wei Lou, and F. Dai. Extended multipoint relays to determine connected dominating sets in manets. *Computers, IEEE Transactions on*, 55(3):334 – 347, march 2006.
- [19] W. Peng and X.-C. Lu. On the reduction of broadcast redundancy in mobile adhoc networks. In *Proc. of Mobihoc'00*, August 2000.
- [20] H. Lim and C. Kim. Flooding in wireless ad hoc networks. *Computer Communications Journal*, 24(3-4):353–363, 2001.
- [21] W. Lou and J. Wu. On reducing broadcast redundancy in ad hoc wireless networks. *IEEE Transactions on Mobile Computing*, 1(2):111–123, 2002.
- [22] M.T. Al Amin, S. Barua, S. Vhaduri, and A. Rahman. Load aware broadcast in mobile ad hoc networks. In *Communications, 2009. ICC '09. IEEE International Conference on*, pages 1 –5, june 2009.
- [23] M. Khabbazzian and V.K. Bhargava. Localized broadcasting with guaranteed delivery and bounded transmission redundancy. *Computers, IEEE Transactions on*, 57(8):1072 –1086, aug. 2008.
- [24] M. Khabbazzian and V.K. Bhargava. Efficient broadcasting in mobile ad hoc networks. *Mobile Computing, IEEE Transactions on*, 8(2):231 –245, feb. 2009.
- [25] Hai Liu, Xiaohua Jia, Peng-Jun Wan, Xinxin Liu, and F.F. Yao. A distributed and efficient flooding scheme using 1-hop information in mobile ad hoc networks. *Parallel and Distributed Systems, IEEE Transactions on*, 18(5):658 –671, may 2007.
- [26] T. Pongthawornkamol, K. Nahrstedt, and Guijun Wang. Hybridcast: A hybrid probabilistic/deterministic approach for adjustable broadcast reliability in mobile wireless ad hoc networks. In *Communications, 2009. ICC '09. IEEE International Conference on*, pages 1 –6, june 2009.
- [27] Li Li, R. Ramjee, M. Buddhikot, and S. Miller. Network coding-based broadcast in mobile ad-hoc networks. pages 1739 –1747, may. 2007.
- [28] Shuhui Yang, Jie Wu, and M. Cardei. Efficient broadcast in manets using network coding and directional antennas. pages 1499 –1507, apr. 2008.
- [29] Brad Williams and Tracy Camp. Comparison of broadcasting techniques for mobile ad hoc networks. In *Proc. of MobiHoc' 02*, pages 194–205, 2002.
- [30] Thomas H. Cormen, Clifford Stein, Ronald L. Rivest, and Charles E. Leiserson. *Introduction to Algorithms*. McGraw-Hill Higher Education, 2001.
- [31] Steven S. Skiena. *The algorithm design manual*. Springer-Verlag New York, Inc., New York, NY, USA, 1998.

- [32] Fei Dai and Jie Wu. Performance analysis of broadcast protocols in ad hoc networks based on self-pruning. *IEEE Transactions on Parallel and Distributed Systems*, 15(11):1027–1040, 2004.
- [33] K. Fall and K. Varadhan. The ns manual. VINT Project, Univ. California, Berkeley, CA, 2001.
- [34] J.-Y. Le Boudec and M. Vojnovic. Perfect simulation and stationarity of a class of mobility models. In *INFOCOM '05: Proceedings of 24th Annual Joint Conference of the IEEE Computer and Communications Societies*. IEEE, 2005.
- [35] Christian Bettstetter, Giovanni Resta, and Paolo Santi. The node distribution of the random waypoint mobility model for wireless ad hoc networks. *IEEE Transactions on Mobile Computing*, 2(3):257–269, 2003.
- [36] Christian Bettstetter. Topology properties of ad hoc networks with random waypoint mobility. *SIGMOBILE Mob. Comput. Commun. Rev.*, 7(3):50–52, 2003.