M.Sc. Engg. Thesis

Optimizing Dominant Pruning Algorithm using Three Hop Neighbor Information

By
Tasmiah Tamzid Anannya

Submitted to
Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
Master of Science in Computer Science and Engineering

Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology (BUET)
Dhaka-1205

February 2018

The thesis titled "**Optimizing Dominant Pruning Algorithm using Three Hop Neighbor Information**", submitted by Tasmiah Tamzid Anannya, Roll No. 1014052061P, Session October 2014, to the Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology (BUET), has been accepted as satisfactory in partial fulfillment of the requirements for the degree of Master of Science in Computer Science and Engineering and approved as to its style and contents. Examination held on February 27, 2018.

Board of Examiners

1. ____________ Dr. A.K.M. Ashikur Rahman Professor Department of CSE, BUET, Dhaka.	Chairman (Supervisor)
2. ____________ Head Dr. M. Sohel Rahman Department of CSE, BUET, Dhaka.	(Ex-Officio)
3. ____________ Dr. Mahmuda Naznin Professor Department of CSE, BUET, Dhaka.	Member
4. ____________ Dr. Sadia Sharmin Assistant Professor Department of CSE, BUET, Dhaka.	Member
5. ____________ Dr. Rajesh Palit Associate Professor Department of Electrical and Computer Engg. North South University Plot #15, Block #B, Bashundhara, Dhaka-1229, Bangladesh.	Member (External)

Candidate's Declaration

This is to certify that the work presented in this thesis entitled "**Optimizing Dominant Pruning Algorithm using Three Hop Neighbor Information**" is the outcome of the investigation carried out by me under the supervision of Professor Dr. A.K.M. Ashikur Rahman in the Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology (BUET), Dhaka. It is also declared that neither this thesis nor any part thereof has been submitted or is being currently submitted anywhere else for the award of any degree or diploma.

Tasmiah Tamzid Anannya
Candidate

Contents

List of Figures

List of Tables

Acknowledgement

First of all, I would like to declare that all the appraisals belong to the **Almighty**. He has blessed me with the capability to continue and complete this research work. I deeply express my gratitude for all His kindness.

I would like to express my deep gratitude to my supervisor, Dr. A.K.M. Ashikur Rahman for introducing me to this interesting field of Wireless Ad Hoc Network and constantly guiding me throughout this whole time. I have learned from him how to carry on a research work and how to write the research. I thank him for his patience hearing of my ideas and reviewing all the draft works, for correcting my flaws, suggesting with new way of thinking and guiding with the proper solution whenever I was in problem.

I am thankful to my parents and my whole family for their endless support and wishes throughout this whole time.

Abstract

Reducing number of forwarding nodes is an important issue to consider when same message needs to be broadcast to all the nodes in a multi-hop ad hoc wireless network. Without proper measures broadcasting may result in many redundant transmissions that increases network traffic and the energy expenses of the nodes. Among many algorithms, the prominent Dominant Pruning (DP) proactive algorithm reduces re-transmissions to a great extent using only 2-hop neighbor information of each node. Partial Dominant Pruning (PDP) and Total Dominant Pruning (TDP) are two variants of DP which use neighborhood information more effectively. According to DP, PDP and TDP, a node selects a subset of nodes from its neighbors as intermediate forwarding nodes based on 2-hop neighborhood information. In this thesis, we propose a new broadcast method based on DP which uses (extended) 3-hop neighborhood information of each node in order to reduce redundancy. Expanding neighborhood knowledge up to 3-hops provides added advantage to the forwarding nodes to detect which nodes are going to be covered by other nodes in the forwarding list of the previous nodes. Thus, redundant transmissions are further reduced using 3-hop neighbor information. Though the new approach requires additional information, the total energy consumed by the entire network and the total number of packet forwarding are much less compared to DP, PDP and TDP.

Chapter 1

Introduction

1.1 Introduction

Wireless mobile hosts can communicate with each other by forming an ad hoc network when structured infrastructure is not available or existing one is inconvenient to use. A wireless ad hoc network is temporarily deployed without any centralized control for emergency purposes like in battlefield, military operations, disaster relief, conferences etc. In these situations, pre-existing infrastructure may not be available to use. Thus, an ad hoc network is formed temporarily in such a way that does not need any central control to communicate among the mobile nodes. As central infrastructure is need not to be installed, ad hoc networks can be constructed at low cost.

In an ad hoc network, each node has to be more reliable than a node in a wired network. Each mobile node in wireless ad hoc network has to act as both host as well as a router. If the destination node is not within the transmission range of the source, then a packet cannot be sent to the destination from source directly. The source needs to rely on intermediate nodes to forward the packet to the destination. Here, the intermediate nodes act as routers to forward the packet and they have to take decisions about the path in which the packet should be forwarded. This scenario is shown in Figure 1.1. Here, node A and C both are in the transmission range of node B. If node A wants to send a message to node C, it needs to rely on node B to forward the packet. In this case, node B will act as router to relay the packet from node A to node C.

As there is no centralized admin node, ad hoc networks face some difficulties that are required to be solved to establish a proper connection among the mobile nodes. Firstly, these

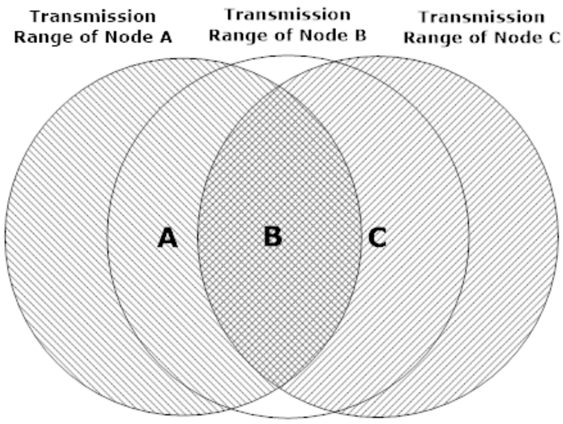

Figure 1.1: Example of wireless network

mobile nodes are resource constrained devices with limited energy and power. If these nodes' power is not used in an organized way, the outcome could be devastating like it could lead to network partition. Therefore, the network gets divided into different sub-networks and these sub-networks get disconnected from one another. Another concern about wireless ad hoc network is, its topology is continuously changing as the node mobility is very high. Despite the mobility, it is often necessary that a piece of information needs to be reached throughout the entire network. Wireless ad hoc networks needs to broadcast messages for various services such as route discovery, periodic data dissemination, erasing an invalid route, locating a node, duplicate IP address detection or even for sending alarm signals in the entire network.

As mentioned earlier, for limited transmission range, a packet may require multiple intermediate nodes from source node to reach to the destination. So, multiple forwarding may be required for a packet to reach to all the nodes in the network. The simplest way to achieve broadcasting is blind flooding. In blind flooding, when a host sends a message, its entire neighborhood will receive the message and upon receiving a packet for the first time, a node rebroadcasts it. Though blind flooding ensures full coverage at high mobility but unfortunately, it results redundant transmissions, high energy consumption and finally leads to broadcast storm problem [1].

Various methods have been proposed to broadcast a message to the network to reduce redundancy. They all use neighborhood information to reduce packet forwarding. These attempts can be broadly classified into two categories- reactive and proactive approach. In reactive approach, upon receiving a packet, a node itself takes decision whether it will forward the packet or not. For instance, with self pruning algorithm, a node attaches its neighbor list in the packet header while forwarding it. When a neighbor of that node receives the packet, it will decide to forward if some of its neighbors are not covered by the previous node. In a proactive scheme, a node selects a subset of nodes from its neighbor as forwarding nodes using neighborhood information and append this information into the packet header. So, when a node receives a packet it knows whether it should forward or not. If it is one of the forwarders, then it will decide its own list of forwarders among its neighbors. One of the effective proactive schemes is Dominant Pruning (DP) algorithm. It uses 2-hop neighborhood information of each node to ensure entire network coverage while reducing redundancy. The forward list is selected from first hop neighbors in such a way that they cover all the nodes within 2-hop. Partial Dominant Pruning (PDP) and Total Dominant Pruning (TDP) are two variations of DP which uses neighborhood information more effectively to produce less number of forwarding nodes. In this thesis, we propose a new approach to reduce redundancy exploiting 3-hop neighborhood information of each node of the network. The key idea behind further optimization is dominant pruning only uses the information about the nodes that have already been covered by the previous node. If a node uses the information about the nodes that are going to be covered by other nodes in the forward list then further optimization is possible and its forward list will be reduced.

1.2 Motivation

Finding the most optimized broadcast tree is NP-complete [2] as the problem is very similar to finding minimum connected dominating set (MCDS) problem in the network. Instead of the optimal solution an approximation is used in many cases. An approximation algorithm for constructing MCDS has been proposed in [3] which require global topology information. This can still be acceptable if the network topology is static. As the nodes in a wireless ad hoc network are mobile hosts, the topology keeps changing. Due to mobility of the nodes, information gathered becomes stale very soon. Thus, it is not desirable for wireless ad hoc network having a centralized admin to collect the global topology information. For this reason, we have to find a solution which will work in a distributed manner quickly for high mobility networks.

Simple blind flooding algorithm does not use any kind of neighbor information or location information or any other information while forwarding a packet. Though blind flooding ensures that the packet reaches to all the nodes of the network, it also creates redundancy, collision and contention–all together known as *broadcast storm*. Dominant Pruning (DP) algorithm tries to reduce redundancy using 2-hop neighbor information of the nodes. A forwarding node attaches its 1-hop neighbor list with the broadcast packet. Each forwarding node also selects which nodes from its 1-hop neighbors should forward the packet further so that all of the 2-hop neighbors of the forwarding node receive the packet. This forward list is also attached to the packet. Upon receiving the packet, a node knows whether it should forward or not. DP obviously outperforms blind flooding, still has a lot of redundancy. For further optimization, two more algorithms namely Partial Dominant Pruning (PDP) and Total Dominant Pruning (TDP) were proposed. PDP and TDP use neighborhood information more efficiently than DP. PDP uses exactly 2-hop information like DP in order to reduce redundancy, whereas TDP uses partial 3-hop neighborhood information of a node and its previous node from where it received the packet for the first time. All these information incur additoinal overhead in the packet header. They both reduce redundancy than DP. TDP outperforms PDP using extended information. So it is easily understandable that the more information a node can afford, the more effectively redundant transmissions could be reduced. Thus, this work attempts to use 3-hop neighbor information of the nodes to cover the entire network at the time of broadcasting a packet and also consumes less energy. The purpose of the work is not to come up with a solution that gives the absolute minimum number of re-transmissions; rather we attempt to provide a solution that will give high effectiveness to reduce redundancy while broadcasting a packet practically. Simply, we seek to minimize broadcast redundancy using 3-hop neighbor information of a node.

1.3 Contributions

The primary contribution of this thesis is to devise an algorithm, *Extended Neighbor Information based Dominant Pruning, ExDP* that utilizes extended neighborhood information to reduce redundancy for broadcasting. The major contributions of this work are enumerated below:

i. Design and development of an algorithm that modifies and improves the already implemented algorithm namely Dominant Pruning so that the same task can be achieved with reduced number of packet forwarding. The improved algorithm overcomes the drawback of Dominant Pruning (DP), Partial Dominant Pruning (PDP) and Total Dominant Pruning (TDP). With the extended neighborhood information improved algorithm, ExDP performs exceedingly better than these three algorithms and it ensures all of the 2-hop neighbors of a forwarding node are covered. As it is an improvement of DP, it inherits all benefits of a proactive algorithm.

ii. A comprehensive comparison of the proposed algorithm with the state-of-the-art algorithms through extensive simulations under different network settings and operating conditions to validate the result.

1.4 Thesis Organization

The rest of the thesis is organized as follows: Chapter 2 presents previous works to reduce redundant messages in a mobile ad hoc wireless network. Chapter 3 defines the terminologies and notations used in this thesis and presents the computation of structural methods, the algorithm and example to present the differences between state-of-the-art methods and the proposed one. Chapter 4 reports experimental results. Finally, Chapter 5 depicts the attainments of this thesis and then concludes the thesis suggesting the future extensions possible.

Chapter 2

Related Work

For different purposes a message needs to be broadcasted to all the nodes of ad hoc network. In recent years, many approaches have been discovered by the researchers to broadcast a message in the network. Some of them are energy-aware schemes, some uses location/neighbor information or history of the routing packet and many other approaches are considered. One of the simplest schemes for broadcasting a message in ad hoc network is blind flooding [1]. Flooding gives the best result for high mobility networks. In pure flooding, when a node receives a packet for the first time, it rebroadcasts it and all of its neighbors receive the packet. In this scheme, a node is compelled to rebroadcast a message seeing it for the first time- which ensures n number of transmissions with a network of n nodes. Thus, it ensures full coverage of the network. According to [1], in a CSMA/CA network, drawbacks of flooding are:

i. **Redundancy:** When a node rebroadcast a message upon seeing it for the first time and all of its neighbors already have the message, it creates redundancy.

ii. **Contention:** After a node sends a message, if many of its neighbors want to rebroadcast it, it will result in serious contention.

iii. **Collision:** No acknowledgement mechanism is used for broadcasting a message because the acknowledgements will also create a lot of traffic around the sender. So chances of occurring collision are high which will led to more damage.

Collectively, these three problems are known as broadcast storm problem. Thus, it is considered inefficient as it results in high redundancy, channel contention and high energy consumption of the network. To solve the problem, authors suggested probabilistic forwarding schemes [4], counter-based schemes [5], distance and/or location aided schemes [6–8], cluster-based schemes [9,10] and neighbor knowledge schemes [2,11–18]. A brief discussion is given below how broadcast storm can be mitigated by these approaches.

2.1 Probabilistic Scheme

When a host receives a message, it rebroadcasts it with predefined probability. A node in the network rebroadcasts a message with *p* probability and does not take any action with *1-p* probability. In this scheme, there is a possibility that packets are dropped and does not reach to a certain part of the network.

2.2 Counter Based Scheme

In Counter Based Schemes, when a host receives a message more than a certain number of times in a certain period, then it decides not to rebroadcast the message. When a message is received for the first time, a timer is started and also noted how many times the same message is received. If the count of the message crosses a certain threshold when the timer ends, then the message is not re-broadcasted by that node. It uses the concept that if a node has received the same message from a certain number of its neighbors, then most likely it is that receiver's 1-hop neighbors have forwarded enough packets to cover all of its 1-hop neighbors. In [5], authors proposed two methods Load-aware Dynamic Probabilistic Flooding (LDPF) and Load-aware Dynamic Counter-based Flooding (LDCF) based on information about the size of its MAC transmission queue, indicating the node's load level. Authors showed that, counter based scheme, LDCF performed better than the probabilistic scheme, LDPF.

2.3 Location Based Scheme

Location based schemes use geographic location of 1-hop neighbors to reduce forwarding nodes. Location can be obtained by GPS (Global Positioning System) receivers or various localized distribution methods if GPS is not available. According to [6], each node selects forwarding nodes from 1-hop neighbors located at the vertices of a regular polygon using location of its 1-hop neighbors. In [7], Optimized Flooding Protocol (OFP) is proposed, in which a node does not need to know locations/addresses of all its neighbors and hence OFP does not impose any bandwidth overhead such as hello messages. Whenever a node transmits a broadcast packet, it also sends the location of the node from which it received the packet and own location. As per [8], Geoflood algorithm is proposed. As mentioned in Geoflood, when a message is forwarded, each forwarding nodes updates the location field with its own location. Upon receiving a packet, the receiver node will measure the quadrant, from which the message is received and a timer is started. Before timer expires, if the message is received from all four quadrants of a circle centered on itself then it is discarded; otherwise forwarded.

2.4 Cluster Based Scheme

In Cluster based approach, group of nodes together form a cluster in the network. Basically, a node with the lowest ID is selected as cluster head. All the neighbors of cluster head are the members of that cluster. If a member node can communicate with another cluster head it is called a gateway member, otherwise it is a non-gateway member. When a message is received from cluster head, the non-gateway members will not rebroadcast but the gateway members will decide whether to forward or not using location/distance/probabilistic approach. As per [9, 10], they used passive clustering method more effectively based on cluster status, neighbor information and other locally gathered information. Information was gathered not by broadcasting periodic hello messages rather using on-going data packets and can construct clusters with partial neighbor knowledge. In both the papers, authors gave gateway selection heuristic to select minimum number of gateways. As per [9], gateways are selected in such a way that there is at most one gateway between two adjacent clusters. If a gateway hears a packet from another gateway for same two clusters then it compares ID of itself with the ID of sender node. If it has lowest ID, it keeps its status as gateway otherwise, it becomes an ordinary member.

2.5 Neighbor Knowledge Methods

Various methods have been proposed to find an optimal flooding tree to reduce redundant flooding using neighbor information of nodes. Neighbor information can be easily gathered by sending periodic beacon messages. Or this information can be found on demand when nodes attach their neighbor list with the broadcast packet. As discussed earlier, in a wireless ad hoc network when a node sends a broadcast packet, all of its neighbors receive the packet. Thus constructing an optimal flooding tree can ensure that if only the nodes belonging into that tree forward the packet, then all the nodes of that network will receive the packet. A flooding tree covers all the nodes in a graph and optimal tree is a tree that minimizes the number of nodes belonging in the tree.

Finding the most optimized broadcast tree is NP-complete [2] as this is similar to finding minimum connected dominating set (MCDS) [19] in the network. If a graph is represented as G(V,E) where V is the set of all nodes in the network and E is the set of edges between the nodes. An edge between two nodes indicates that they both are within their transmission range. Finding an MCDS of G(V,E) is to find minimum connected subset S of V where each node of V-S is adjacent to at least one node in S. That means, a node in the network is a member of either S or V-S.

An approximation algorithm for constructing MCDS works in a greedy approach. At first, all the nodes in the network are colored white. Then the node with the maximum node degree is selected and colored as black, while all of its adjacent nodes are colored as grey. From the grey nodes, select the node having the maximum white neighbors. In same way, color the selected grey node black and all its neighbors grey. This process runs until no white colored node exists. The resultant set of black nodes is the approximation of MCDS. This method is inappropriate for wireless ad hoc network as it requires global topology information. Obtaining the whole topology in a wireless ad hoc network is like “shooting your own foot” because of the mobility of the hosts. Thus, some kinds of heuristics are needed to be developed which can work using little topology information. Various heuristics have been proposed to minimize the flooding. Mainly, all mechanisms can be divided into two categories which are given below:

2.5.1 Reactive approach

In this approach, upon receiving a packet, a receiver decides whether to forward the packet or not. Some of the reactive approaches are given below in short:

i. **Self Pruning (SP) Algorithm:** In [2], Lim and Kim proposed a reactive approach named Self Pruning (SP). In SP, a node, u attaches its neighbor list, $N(u)$ while forwarding a packet. Another node v, in the transmission range of node u when receives the packet compares its own neighbor list, $N(v)$ with $N(u)$. If there are new nodes that v can cover then its forwards the packet. Basically, if $N(v) - N(u) \neq \emptyset$, then it forwards. However, since there could be several receivers of u those cover the same node w and the nodes take decision without coordinating among themselves, there will remain a lot of redundant re-transmissions.

ii. **Scalable Broadcast Algorithm (SBA):** In [20], another algorithm named Scalable Broadcast Algorithm is proposed. This proposed idea is similar to Self Pruning. Like SP, instead of considering only the previous transmission, a node does not rebroadcast the packet if all of its neighbors have received the packet from previous transmissions. It is achieved by delaying the rebroadcast for a random period. In this waiting period, when a node receives the duplicate messages from other nodes, it continues comparing its own neighbor list with the neighbor list of later sender nodes and updates the covering set.

iii. **Improved Self Pruning:** In Improved self pruning [21], each node makes decision whether to forward or not based on 3-hop neighbor information and performs much better than traditional Self Pruning.

2.5.2 Proactive approach

While transmitting a packet, the transmitter node decided which of its neighbor nodes should forward the packet. The forwarder list is attached in the header of the broadcast packet. Then receiving a packet, a node knows its role. A receiving node who is requested to forward the packet again determines the forwarder list from its neighbors similarly; otherwise, does not construct any forward list and does not rebroadcast. The flooding ends when there is no more node to transfer anymore. Recently different types of proactive approaches have been suggested by researchers. Some of the remarkable approaches of proactive method are given below:

i. **Dominant Pruning (DP):** Lim and Kim proposed another algorithm other than SP in [2], which is Dominant Pruning. Among bind flooding, self pruning and dominant pruning, the later one performs much better than others. It reduces redundancy considering larger region while making decision. When a node, v receives a packet from another node, u, and it is in the forward list of u then it selects minimum number of nodes from its 1-hop neighbors deducting the neighbors of u, $N(v) - N(u)$ to cover all the nodes of the set $U_v = N(N(v)) - N(v) - N(u)$. $N(N(v))$ is all the 2-hop neighbors of v. $N(v)$ is discarded as they will receive when v will forward and u has already forwarded, so $N(u)$ is also covered by u. By selecting nodes cleverly, redundancy is reduced. Simulation shows DP outperforms both blind flooding and Self Pruning.

ii. **Partial Dominant Pruning (PDP):** Authors in [11], suggested two heuristics of dominant pruning- Partial Dominant Pruning and Total Dominant Pruning. Partial Dominant Pruning uses 2-hop neighbor information more effectively. Other than deducting $N(v)$ and $N(u)$ from $N(N(v))$, PDP also deducts the neighbor of common neighbors of node u and v. This reduction of sets reduces number of forward list than DP.

iii. **Total Dominant Pruning (TDP):** TDP requires 2-hop neighbors of the immediate sender node to be piggybacked with the broadcast packet. Thus, a node v receiving a packet from node u also deducts $N(N(u))$ from $N(N(v))$. Here, partial 3-hop neighbor connectivity is actually used to make the decision.

iv. **Enhanced Partial Dominant Pruning (EPDP):** According to [15], EPDP is an extended version of PDP. Only difference between PDP and EPDP is, EPDP introduces a delay before forwarding a received broadcast packet. It takes advantage of the fact, that the same node may hear the same packet several times from its neighbors. Whenever the node receives the same packet from another neighbor, it uses the overheard information and keeps updating the U set, it needs to cover according to the newly received packet. After the timer is expired, if U set is empty, the node does not forward. Even if U is not empty, it is smaller than PDP for many nodes. The defer time is selected according to the position of the node in the forward list. The nodes with larger coverage area will be in the former position in forward list. Thus, the nodes in the higher position in the list are given lower delays and the nodes lower in the forwarding list are given higher delays. This solution brings better performance than DP and PDP. Basically, EPDP is combination of reactive and proactive approach. As the nodes in the forward list from the immediate sender defers its re-transmission and after defer time the node itself decides whether to rebroadcast the message or not.

v. **Enhanced Dominant Pruning (EDP):** Authors in [14] showed that, DP considering only $N(v) - N(u)$ for forwarding sometimes leads to incorrect result for particular topologies. As stated in this paper, the nodes shared by sender node and receiver node are still candidate for forwarding as node v may have some 2-hop neighbors that can only be covered by those shared nodes. As the forward list is attached with the broadcast packet, v can easily remove those nodes from shared nodes that have already been chosen by the immediate sender node. This method requires more information than DP. While forwarding a packet, a node needs to attach the forward list it created from its 1-hop neighbors and also attaches its sender's forward list. Using this information EDP results in fewer broadcasts than DP. Authors proposed EDP for route discovery and integrated with AODV and shows it performs better than standard AODV. However, even if the shared nodes are not considered as the candidate nodes, the authors in [2] ensured that DP covers all the nodes in the network.

vi. **Multipoint Relaying (MPR):** Similar to DP, MPR [12] uses 2-hop neighbor information to reduce broadcast redundancy. There are several differences between these two. At first, when a node v receives a broadcast packet from node u, its U set-the set it needs to cover is $N(N(v))$, all 2-hop away neighbors of node v. It does not deduct any other information from $N(N(v))$ like DP. Thus, the set need to cover is much larger than DP. There are some nodes in U set which can only be covered by only one node in 1-hop neighbors of node v. To optimize the set of forwarding nodes, these 1-hop neighbors are added to MPR_v set. Now, for the remaining nodes (if exists) in $N(N(v))$ which are not covered by MPR_v set, the greedy set cover algorithm is used.

vii. **AD Hoc Broadcast Protocol (AHBP):** AHBP [22] uses 2-hop neighbor information as well as the history about the route the packet has traversed. In AHBP, not only the forward list is piggybacked in the packet but also the packet carries the whole path it has traversed from source. Undoubtedly, this route information creates additional extra overhead and increases the size of the packet. The set that needs to be covered, U can be potentially reduced by removing the nodes that the packet visited and their 1-hop neighbors.

viii. **Relative Degree Adaptive Broadcast (RDAB)** According to [23], the relative degree of a node is used to take decision whether this node should be included in forward list or not. The relative degree of a node is denoted as the neighbor nodes of a node, those have not received the broadcast packet from earlier iterations. It may be equal or less than the actual node degree of the node. A node selects from its 1-hop neighbors as forwarder whose relative degree is the highest and the relative degree of the nodes are updated after each iteration. The node with highest relative is selected until all the 2-hop nodes are covered. This method is very similar to DP.

ix. **k-Shortest Path Pruning (k-SPP):** Dhan and Rieck proposed another method called k-Shortest Path Pruning (k-SPP) algorithm in [17]. In this method, each node must maintain an awareness of its $k + 1$ hop neighbors and also the broadcast packet carries IDs of the most recent k nodes it has visited. While deciding forward list node v [received packet from node u and is in the forward list of u] check whether any of its 2-hop neighbors are also 2-hop neighbor of another node in the forward list of u. If v 's ID is less than the other node that will cover any certain 2-hop away node of v, then v will deduct that certain node from its U set . Again, if node v does not lie in the shortest path from a node (the node which is the first node in the history path carried by the packet) to any 2-hop neighbor of v, then it will also not consider that node to cover. In k-SPP, a node that has already forwarded the message may forward it again, if it receives again via shorter path.

x. **Smart Gossip:** In [24], authors suggested a technique for communicating among nodes in a wireless sensor network name ”Smart Gossip”. It is actually a combination of probabilistic approach and neighbor connectivity approach. In this approach, each node forwards a message with a probability , p_{gossip} and selects the value of p_{gossip} by overhearing the messages by neighbors. The importance of a node increases, when other nodes heavily depend on this node to receive a packet and it has to forward packets with high probability. in more dense part of the network, sensors may use lower probability value. This gossip protocols can be categorized in to two approaches- static approach and adaptive approach. In static approach, all the nodes use a single p_{gossip} value and thus it has to be high to make sure that the packet is reached to the sparse parts of the network. As the probability should be update over time depending on the link loss of the sensors and mobility,s static approaches are not suitable for wireless networks. Therefore, the authors in this paper suggested a new adaptive approach where the information about dependencies among nodes is extracted by overhearing messages from the neighbors. Like, when a node X depends on some nodes to receive a message, they become the parents of X, $parent(X)$ and when other nodes depend on X, they are identified as children of X, $child(X)$. There may be some other neighbor nodes of X, neither they depend on X, nor X depends on them. They are identified as siblings of X, $sibling(X)$ by node X. Initially, when dependency is not known, message is forwarded with probability 1. Over time, these dependencies are learned and modified. Each node computes it's parent's gossip probability and updates a specified field with the computed value when forwards a message. Upon receiving the computed value by each child, the parent node chooses the maximum value and forwards the message with this p_{gossip} value. If no message is received from a node for a specified duration, then the node is deleted from all dependencies and the whole parent-child relationship is revised and updated. Thus, this approach also deals with the topology changes and node failure. However, it will still result in a lot of redundancy as the nodes have to spread the messages to announce it's present to it's parents even if it does not have any child node depending on it.

xi. **Three-Hop Horizon Pruning (THP):** Authors of the paper [25], build a two hop connected dominating set such that any node in the set is two hop away from some node of the set. They also used 3-hop neighborhood information but unlike DP, while forwarding a packet a node covers forwarders so that all the nodes within 3-hop away gets the packet. Here, a node attaches it's 1-hop dominating set along with 1-hop neighbors while forwarding HELLO packets. Thus, a node gets knowledge how a message needs to be propagated so that the nodes within 3-hop neighbor range receive the packet, even though the nodes are unknown.

2.6 Analytical summary:

The following can be observed by the proposed systems:

i. Different systems adopt different methods from probabilistic-based, distance-based, location-based, and cluster-based or combination of these based on their demand. Basically, these all are some kind of probabilistic scheme, so do not guarantee to cover the whole network.

ii. Most of the systems perform tremendously better than pure flooding and thus reduce broadcasting redundant packets and also saves power consumption.

iii. Some systems need route history to be piggybacked in the packet along with neighbor information. These obviously increase the packet size after a certain period of time and create additional overhead to broadcast the packet. It also makes the algorithm complex and time consuming.

iv. DP, PDP and TDP exceedingly outperform blind flooding and SP in cost of little neighbor information and also keep the whole procedure simple.

Our proposed technique, ExDP optimizes DP by exploiting neighbor information of nodes from 2-hop to 3-hop while broadcasting a packet in the network. Using the complete 3-hop neighbor information while taking decisions, the redundant transmissions are reduced to a great extent in cost of little additional overhead. This will not be a big problem as the total broadcast is reduced immensely.

Chapter 3

Methodology

This thesis work proposes an algorithm called, Extended neighbor information based dominant pruning, ExDP which improves the prominent broadcast method dominant pruning, DP of wireless ad hoc network by using 3-hop neighbor information of all the nodes in the network. The whole methodology of ExDP and other related approaches are explained with example in this chapter.

3.1 Network Model

A wireless network can also be represented as a graph like a wired network. Each mobile host is modeled as a vertex in the graph and an edge between two nodes indicates they are within the transmission range of each other. As the hosts of a wireless network are mobile nodes, due to their mobility the graph changes from time to time.

We model a wireless ad hoc network as a simple undirected graph $G(V, E)$ where V is set of all vertices (mobile hosts) in the network and E means set of all edges in the network. An edge, $e(v_i, v_j)$ between two nodes v_i and v_j will be included in the graph if $d(v_i, v_j) <= r_i$ and $r_i = r_j$, where r_i and r_j is the transmission range of node v_i and node v_j respectively and $d(v_i, v_j)$ is the Euclidean distance between node v_i and node v_j.

Figure 1.1 shows an example of wireless ad hoc network consisting of three nodes A, B and C. Here, A and C both are in transmission range of B. Again, B is in transmission range of A and C. The circle around a node indicates is transmission range. The nodes within the circle are considered as the neighbor hosts of that node. That means, when a node forward a packet, all the nodes within this transmission range receives the packet. As, node A and node B both are within each other's transmission range, they can communicate with each other and in the network graph there will be an edge between them. The same case also applies for node B and node C. Figure 3.1 is the conversion of the graph of the network from Figure 1.1.

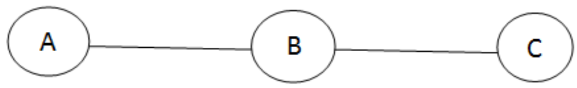

Figure 3.1: Representation of ad hoc network

In a real life wireless network, some edges between nodes may be unidirectional because of the variation of transmission range of nodes. However, we assumed all the nodes have the same transmission range, thus all the edges in our graph is bidirectional. If node v_i can communicate with node v_j, then v_j can also communicate with v_i.

3.2 Assumptions

For simulation, we made some assumptions:

1. All the hosts have omni-directional antennas that means signals are received or transmitted from all the directions.
2. All the nodes in the network have the same transmission range.

3.3 Definitions

Let us assume v is an arbitrary node of the network and it has received a broadcast packet from node u for forwarding. Node v will create a forward list and append the list in the packet header before rebroadcasting. Although the creation of the forward list would vary from algorithm to algorithm, in general all algorithms would use the following notations:

1. **N(v):** N(v) is the set of all 1-hop neighbors of node v. The nodes that are in the transmission range of v are member of this set. Note that, node v itself also is a member of this set, $v \in N(v)$.

2. **N(N(v)):** N(N(v)) is the set of all nodes within 2-hop of node v. Obviously, $\{v\} \subseteq N(v) \subseteq N(N(v))$.

3. **N(N(N(v))):** All the nodes that are within 3-hop from node v. $N(N(v)) \subseteq N(N(N(v)))$.

4. **N(N(v))-N(v):** The nodes that are exactly 2-hop away from v are included in this set.

5. **F(v):** The list of 1-hop neighbors of v that are selected for forwarding by node v. $F(v) \subseteq N(v)$.

6. **u:** The previous hop node of v. Node u has selected node v to forward the message. So, $v \in F(u)$.

7. **$\mathbf{B_v}$:** The set of one hop neighbors of node v that are eligible to be included in the forwarding list $F(v)$. When node v receives a packet from node u and $v \in F(u)$, it selects its own forward list. Node u and node v may have some common neighbors, so while selecting forwarding nodes, node v does not need to consider those common neighbors as they were already considered by node u. Thus, $B_v = N(v) - N(u)$ and node v selects forwarding nodes from B_v set. That means, $F(v) \subseteq B_v$.

8. **$\mathbf{U_v}$:** U_v is the set of nodes that need to be covered by using nodes from B_v while v creates its forwarding list.

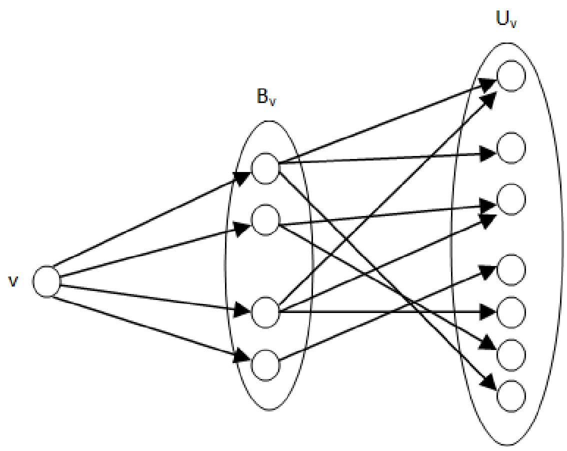

Figure 3.2: The broadcasting problem mapped to the set cover problem

The problem of optimizing redundancy can be easily mapped to a set cover problem. Upon receiving a packet from node u, if node v is in $F(u)$, node v creates its own B_v and U_v set.

Then node v selects some nodes from B_v set to cover all the nodes that belong to U_v set. The whole scenario can be mapped as a set cover problem as indicated in Figure 3.2. Node v needs to choose some nodes from B_v in such a way that the nodes in U_v are neighbor of at least one node of B_v set. Thus, it is clearly evident that, the more the size of U_v reduces, the less number of nodes from B_v is selected. That is why, B_v set is similar in all the algorithms–DP, PDP, TDP and the proposed one but all these algorithms cut down U_v set differently in order to reduce the size of U_v set. Normally, PDP deducts more nodes from U_v set than DP and TDP deducts even more nodes compared to PDP. The proposed algorithm has the least number of nodes in the U_v set than others and thus, it selects least number of nodes from B_v set while creating forward list.

3.4 Considered Algorithms

The proposed Extended Neighbor Information Based DP dubbed as "ExDP" optimizes the Dominant Pruning (DP) algorithm for reducing redundancy to avoid broadcast storm problem. Thus, before proceeding to the proposed methodology, at first we provide a short description of DP along with PDP and TDP for better understanding of the readers next.

3.4.1 Dominant Pruning(DP) Algorithm

In DP a node does not take decision whether it should broadcast the packet or not. To avoid redundancy of coverage of forwarding nodes, the previous node makes such decision. If a node is selected by the previous hop as forwarder it forwards the packet; otherwise it discards the packet after retrieving the data. Moreover, the forwarding node also selects forwarders from its 1-hop neighbors. The selection methodology is as follows.

Suppose, a node v, receives a packet from node u (source) and is selected as a forwarder. Now, it selects some nodes from $N(v)$ to cover the nodes in $N(N(v))$. Node u being the source, among the nodes in $N(N(v))$, v and $N(u)$ have already received the packet. $N(v)$ will receive the packet when v will forward. Thus, v will select forwarding list to cover only the nodes in $U_{\mathrm{v}} = N(N(v)) - N(v) - N(u)$ from $B_{\mathrm{v}} = N(v) - N(u)$, as nodes in $N(u)$ have already been considered by node u. Node v selects forward list, $F(v) = \{f_1, f_2, \ldots f_i\}$ such that $F(v) \subseteq B_v$. This forward list is piggybacked with the broadcast packet. It is basically a classical set cover problem. We need to cover the nodes in U_{v} set by the nodes in the B_{v} set.

As finding a minimum set of forwarding node is NP-Complete, here a greedy set cover algorithm is used. In the selection process node v_{k} is selected with the maximum number of uncovered neighbor nodes. The algorithm for DP is given below:

Dominant Pruning (DP) Algorithm [11]:

1. Node v uses $N(N(v))$, $N(u)$,and $N(v)$ to obtain $U_v = N(N(v)) - N(u) - N(v)$ and $B_v = N(v) - N(u)$.

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

Selection Process to find forward list, F: [2]

1. Let $F = \emptyset$, $K = \{S_1, S_2, ..., S_n\}$ where $S_k = N(v_k) \cap U(1 \leq k \leq n)$, $Z = \emptyset$.

2. Find the set S_k whose size is maximal in the set K.

3. $F = F \cup v_k$, $Z = Z \cup S_k$, $K = K - S_k$,$S_l = S_l - S_k$ for all $S_l \in K$.

4. If $Z = U$, then complete the algorithm.

5. Otherwise, repeat from 2 again.

The same selection process mentioned above have been used by PDP, TDP and the proposed ExDP.

3.4.2 Partial Dominant Pruning (PDP) Algorithm

In [11], authors proposed PDP which improves performance than DP using the same information as DP more efficiently. Based on the 2-hop neighbor connectivity information PDP is able to deduce more nodes from $U_v = N(N(v)) - N(v) - N(u)$. Clearly, the more reduction of the set is made the more reduction in the number of forwarding nodes from B_{v} is possible. The nodes that can be excluded from set U_{v} are the neighbors of $N(u) \cap N(v)$. As these nodes are exactly 2-hop away from the previous node u, qand node u has already selected nodes from $N(u)$ to cover them. These nodes are included in P set, such that, $P = N(N(u) \cap N(v))$. Therefore, U_{v} set is redefined in PDP as $U_v = N(N(v)) - N(v) - N(u) - P$.

Partial Dominant Pruning (PDP) Algorithm [11]:

1. Node v uses $N(N(v))$, $N(u)$,and $N(v)$ to obtain $P = N(N(u) \cap N(v))$, $U_v = N(N(v)) - N(v) - N(u) - P$, and $B_v = N(v) - N(u)$.

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

PDP algorithm does not increase any extra overhead than DP or does not need any additional information. Still it gives outstanding result by outperforming DP. Difference with DP is that the forwarding nodes need to compute additional P Set.

3.4.3 Total Dominant Pruning (TDP) Algorithm

TDP requires a little bit more information than DP and PDP. As we are assuming u is sending the packet to node v, it means, u has already computed $F(u)$ set in a way that all the nodes in $N(N(u))$ is covered. If node v can receive the $N(N(u))$ information with the broadcast packet, it will be able to reduce the size of U_{v} set. Therefore, the 2-hop away neighbor set U_{v} that needs to be covered by $F(v)$ becomes $N(N(v)) - N(N(u))$. Node v is using 2-hop neighbor information of its 1-hop away node, only 3-hop neighbor information it requires is from node u which is already piggybacked in the packet header. We can see, actually partial 3-hop node information is used to reduce the size of U_{v}.

Total Dominant Pruning (TDP) Algorithm [11]:

1. Node v uses $N(N(v))$, $N(N(u))$, $N(u)$,and $N(v)$ to obtain $U_v = N(N(v)) - N(N(u))$ and $B_v = N(v) - N(u)$.

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

The correctness of excluding $N(N(u))$ from $N(N(v))$ has been shown in [11]. This is provided in the following *Lemma 1*.

Lemma 1: If a node $w \in N(N(v))$ is also in $N(N(u))$, then node w can be excluded from U_{v}.

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

The fact that forward nodes can be selected from B to cover U in TDP algorithm is shown in the following *Lemma 2*.

Lemma 2: Let $U_v = N(N(v)) - N(N(u))$ and $B_v = N(v) - N(u)$,then $U \subseteq N(B)$.

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

TDP requires extra overhead then DP and PDP. However, the outputs of TDP and PDP are very close to each other. Using additional information than PDP, TDP does not help much to reduce redundancy. The key idea is that the dominant pruning reduces number of broadcasts by using the nodes that are covered *only* by the previous node. However, further optimization is possible if the node takes into account all the nodes mentioned in the forwarding list instead of just only the *previous* node. Because, then it can reduce its 2-hop uncovered nodes' list by excluding those nodes that are going to be covered by the other forwarding nodes in the subsequent broadcasts. Thus, the node's uncovered node list will eventually become much reduced compared to the dominant pruning (DP) algorithm - creating a scope for further optimization.

3.5 The Extended Neighbor Information Based Dominant Pruning (ExDP) Algorithm

The proposed method dubbed as Extended Neighbor Information Based Dominant Pruning (ExDP) falls into the category of proactive broadcast scheme which reduces redundant broadcast packets by using 3-hop neighbor connectivity information of a node. Unlike TDP, it does not require 2-hop neighbor information of the previous hop to be piggybacked with the broadcast packet. Rather each node requires to have full 3-hop neighbor connectivity information. Because in order to determine the nodes that will be covered by the nodes in a forwarding list, an intermediate forwarding node needs to collect its neighborhood information up to 3-hop. This can be achieved by forwarding the periodic beacon messages with a time-to-live (TTL) field set to three. Note that, in wireless networks a node usually sends beacon messages to all its neighbors to indicate its presence. We just need to propagate it up to three hops instead of one hop. There will be some delay while propagating HELLO packets up to three hop neighbors. In high mobility scenario, this information may be stale by the time packet reaches to the source. However, the problem of dealing with stale information is not the focus of this thesis paper. The whole procedure of using 3-hop neighbor information to reduce redundant packets is described next.

The objective of ExDP algorithm is to reduce redundant broadcast packets which are necessary to flood the network with same guarantee as Dominant Pruning (DP) algorithm. Let us assume, in a network all the nodes have knowledge up to their 3-hop neighbor nodes. As mentioned earlier, set U_{v} plays an important role in DP. If the size of U_{v} is less, then the number of forwarding list also becomes fewer. This size of U_{v} is less in PDP and TDP compared to DP. Actually, $(size\ of\ (U_v))_{TDP} \leq (size\ of\ (U_v))_{PDP} \leq (size\ of\ (U_v))_{DP}$. In ExDP, each node uses its complete 3-hop neighbor information to reduce the size of U_v set further.

Suppose, a node u (source) initiates a broadcast packet and all the nodes in the transmission area of node u receive the packet. Node u also selects some of its 1-hop neighbors as forwarding nodes and the forward list is attached with the broadcast packet. Assume, node v and node w are such nodes in the forwarding list, i.e., $v \in F(u)$ and $w \in F(u)$. As each node has complete 3-hop neighbor information, a forwarding node v has knowledge about the 2-hop away neighbors of node u, which is denoted as $N(N(u))$. Now, using 3-hop neighbor information node v can deduct $N(N(u))$ from its 2-hop away neighbors $N(N(v))$ in U_{v} just like TDP,

$$U_v = N(N(v)) - N(N(u)) \tag{3.1}$$

Now, more optimization is possible than TDP because there might be some nodes which will be included in U_{v} that are neighbor of more than one node in the forward list of u. In the following Figure 3.4 and 3.3, such scenarios are presented. For Figure 3.3, node v and w both will include node x in their uncovered set as node s is exactly 2-hop away from them. Node v and w both will think how node x can be covered and they both will ask node y to forward to cover node x. Therefore, it is clearly seen that all the forwarding nodes include the common two-hop away neighbors of their common neighbors.

Node v and node w is the neighbor of node u. Node u will ask both v and w to forward as node y is covered by only node v and node z is covered by only node w for Figure 3.4. Therefore, both node v and w will forward the broadcast packet received from the source node

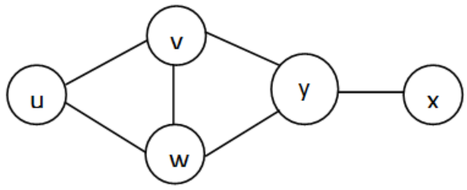

Figure 3.3: A sample network

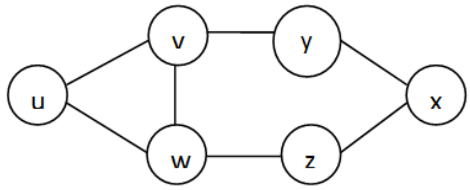

Figure 3.4: A sample network

u. Here, node x is a common neighbor of the un-common neighbors of two forwarding nodes v and w. Now, y and z need to construct their own forward list from their 1-hop neighbors. Although node y will be asked twice to forward in case of DP, it will forward only once so, no redundant transmission occurs here. Still, both node v and w does not need to include x in their uncovered list.

Thus, both v and w will include node x in their uncovered set. In DP, node v will ask node y to forward to cover its 2-hop away node x. Similarly, node w will also ask node z to forward the message to cover its 2-hop away node x. So, both node y and z will forward the message and both will cover node x. It is clearly seen, that only one re-transmission is enough (either node y or node z) to cover node x. It happens because in DP, the forwarding nodes forward packet without any co-ordination among them. This redundant transmission is avoided by using the proposed ExDP method of this thesis. To achieve this, the nodes in the forward list does not need to have any extra communication among themselves, the nodes just need to have 3-hop neighbor information. Using 3-hop neighbor connectivity knowledge, either node v or w would forward the packet, not both.

If node v (which is in the forward list of u) uses the information about the nodes that are going to be covered by other nodes in forward list of u, then it's forward list is reduced more than dominant pruning (DP). Thus, if another node w which is also in the forward list of, $F(u)$ is neighbor of node v (and ID of node w is smaller than ID of node v), then using 3-hop neighbor information v can easily determine the $N(N(w))$ as $N(N(w)) \subseteq N(N(N(v)))$ Thus, v can come to a decision that $N(N(w))$ are going to be covered by w when it forwards and it can subtract $N(N(w))$ from its uncovered set U_{v}. Then, node v will not ask node y to cover, rather node w will include node z in its forward list to cover node x. Thus, only one node between y and z will forward the packet.

Figure 3.5 highlights the overall idea. From the figure, it is easy to see that if there exists a node x which is exactly 2-hop away from both v and w [if, $x \in N(N(v)) - N(v)$ and $x \in N(N(w)) - N(w)$], then either node v or node w will include node x in U list. Which one from node v and w will cover node x depends on their ID. The node with the smaller ID will include node x in its U list. However, if ID of node w is not smaller than ID of node v, then v will not deduct $N(N(w))$ from U_{v} but w will deduct $N(N(v))$ from U_{w}. Then, according to Figure 3.4, node v will ask node y to forward to cover node x.

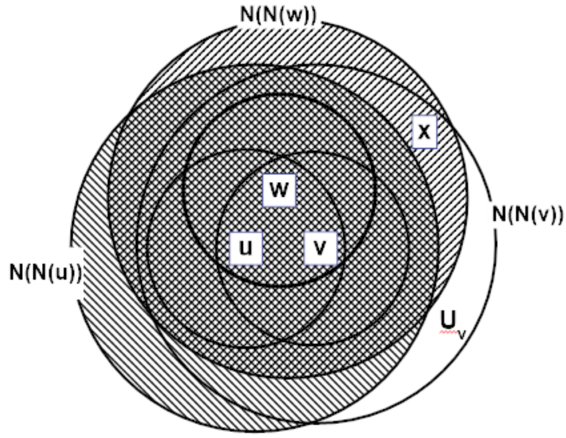

Figure 3.5: Illustrating ExDP algorithm

Formally, let, $F(u) = \{f_1, f_2 \ldots ., f_i\}$, then another set Q_{v} of node v [$v \in F(u)$] can be calculated by the following equation-

$$Q_v = \bigcup_{i=1}^{|F(u)|} N(N(f_i))\ where\ ID(f_i) < ID(v)\ and\ f_i \in F(u) \cap N(v) \tag{3.2}$$

Thus, the final equation of U_{v} becomes:

$$U_v = N(N(v)) - N(N(u)) - Q_v \tag{3.3}$$

Algorithm of proposed Extended Neighbor Information based Dominant Pruning (ExDP) operates as follows:

Extended Neighbor Information based Dominant Pruning (ExDP) Algorithm:

1. Node v has received a packet from node u. If it has not seen the packet and v is in the forward list of u, $F(u)$ then it proceeds to step 2, otherwise discards it.

2. Let $F(u) = \{f_1, f_2 \ldots .., f_m\}$, $v \in F(u)$ and $Q_v = \emptyset$.

3. For each $f_i \in F(u)$, if ID of f_{i} is smaller than ID of node v and $f_i \in N(v)$, then node v constructs Q_{v} as follows,

$$Q_v = Q_v \bigcup N(N(f_i)) \tag{3.4}$$

4. Now node v uses $N(N(v))$, $N(N(u))$, $N(u)$, and Q_{v} to obtain U_{v} and B_{v}.

$$U_v = N(N(v)) - N(N(u)) - Q_v \tag{3.5}$$

$$B_v = N(v) - N(u) \tag{3.6}$$

5. Node v calls the selection process to determine $F(v)$ same as DP.

This algorithm is triggered when a packet is received by a node v from node u. If node v has already seen the message then it discards it. Otherwise, if it has received the message for the first time and is in the forward list of u, it creates its own forward list. The other forwarding nodes selected by u are also known to node v as this list is attached with the broadcast packet. Thus, in step 3 node v selects the nodes from the forward list of u which are neighbors of v and also has lower ID than node v. Using 3-hop neighbor information node v calculates Q_{v} with their 2-hop neighbors. In step 4, node v removes these nodes from its U_{v} set. Then it goes back to the basic set cover problem and using greedy set cover approach selects its own forward list, $F(v)$.

3.6 Correctness of the Proposed Method

The correctness of excluding $N(N(f_{\text{i}}))$ from $N(N(v))$, where $F(u) = \{f_1, f_2 \ldots .., f_m\}$ in case of $f_i \in (F(u) \cap N(v))$ and $ID(f_i) < ID(v)$, is shown in the following *Theorem 1*.

Theorem 1: If a node $x \in N(N(v))$ and also $x \in N(N(w))$, then x can be excluded from U_{v}. [Condition: v and $w \in F(u)$, $w \in N(v)$ and $ID(w) < ID(v)$]

Proof: U_{v} set contains the nodes which needs to be covered by v and exactly 2-hop away from v. If a node x is in $N(N(v))$ and x is also in $N(N(w))$, then three cases may happen:

i. $x \in N(w)$

ii. $x \in N(v)$

iii. x is exactly 2-hop away from both v and w and not in $N(N(u))$, $x \in N(N(v)) - N(N(u))$ and $x \in N(N(w)) - N(N(u))$.

For cases (i) and (ii), x can easily be excluded from U_v as it will receive the packet when node v and node w will forward the packet. It is confirmed that, both node v and w will forward because they both are in the forward list of previous hop which is node u. For case (iii), in case of TDP, the node x will be covered by both v and w. Thus, v can exclude node x from U_v as x will be covered by node w. [In case of $ID(w) < ID(v)$]

Now, the nodes selected from B_v set can cover the nodes in U_v is shown in *Theorem 2*:

Theorem 2: Let $U_v = N(N(v)) - N(N(u)) - Q_v$ and $B_v = N(v) - N(u)$ where $Q_v = Q_v \bigcup N(N(f_i))$, then $U_v \subseteq N(B)$.

Proof: In *Lemma 2*, it is shown that, using the fact that $N(X) - N(Y) \subseteq N(X - Y)$, $N(B) = N(N(v) - N(u))$ can cover $U_v = N(N(v)) - N(N(u))$. Here, $Q_v = N(N(N(v) \cap N(u)))$,when ID of the nodes in $N(v) \cap N(u)$ are smaller than ID of node v. Therefore, it is clearly seen that, when Q_v is subtracted from $U_v = N(N(v)) - N(N(u))$, $U_v(ExDP) \subseteq U_v(TDP)$. So, $N(B)$ can cover newly constructed U_v.

3.7 Example

Let us provide an example to demonstrate the operation of DP, PDP, and TDP along with the optimized algorithm to show the differences and how ExDP improves performance than others. For this, a sample network in Figure 3.6 is considered whose neighbor list is illustrated in Table 3.1. Even though 2-hop neighbors can be derived from one-hop neighbor information, both the information is listed in the table to facilitate our explanation. Note that, a node v is included in the list of $N(v)$.

Figure 3.6 is a network with 12 nodes. Suppose, node 2 has initiated a broadcast packet. Now the list of forwarding list for DP, PDP, TDP and proposed method will be shown to indicate the differences among them. Let us begin with DP, when node 2 broadcasts a packet, it constructs-

$$U_2 = N(N(2)) - N(2) = \{4, 7, 9, 11\} \tag{3.7}$$

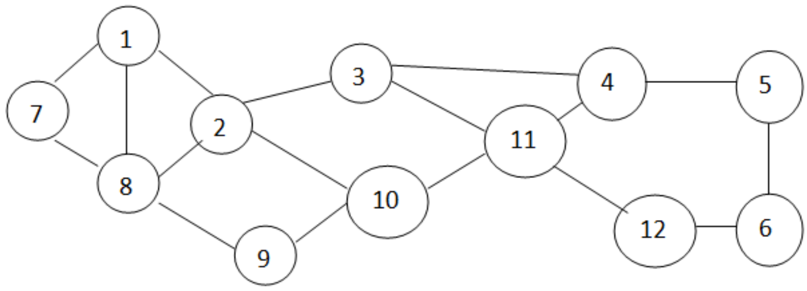

Figure 3.6: A sample network of 12 nodes with source node 2 to show the improvements of proposed method

Table 3.1: 1-hop and 2-hop Neighborhood Information of Each Node

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

$$B_2 = N(2) = \{1, 3, 8, 10\} \tag{3.8}$$

So, node 2 must select among the four nodes in B_2 to cover the nodes in U_2. Consequently, the forward list of node 2 becomes,

$$F(2) = \{3, 8\} \tag{3.9}$$

Node 3 and 8 are in the forward list of node 2 among the neighbors node 2; therefore, node 3 and node 8 will re-broadcast the packet. Now, if node 3 re-broadcasts it determines its U_3 and B_3 list as follows:

$$U_3 = N(N(3)) - N(3) - N(2) = \{5, 12\} \tag{3.10}$$

$$B_3 = N(3) - N(2) = \{4, 11\} \tag{3.11}$$

Table 3.2: The DP Algorithm

u	v	U_v	B_v	F(v)
∅	2	4,7,9,11	1,3,8,10	3,8
2	3	5,12	4,11	4,11
2	8	3	7,9	[ ]
3	4	6,10,12	5	5
3	11	5,6,9	10,12	10,12
4	5	12	6	6
11	10	1,8	2,9	2
11	12	5	6	6
5	6	11	12	12

Thus, the forward list becomes:

$$F(3) = \{4, 11\} \tag{3.12}$$

Similarly, for node 8, we have,

$$U_8 = \emptyset \tag{3.13}$$

$$B_8 = \{7, 9\} \tag{3.14}$$

As there is no node in U_8 set, so node 8 does not need to ask any of its neighbors to forward; therefore,

$$F(8) = \emptyset \tag{3.15}$$

This process continues until all the nodes in the network have received at least one copy of the packet. The detail construction of U, B and F set for DP is shown in Table 3.2. It takes total 9 nodes to forward the packet to cover the whole network, including the original broadcast initiated by node 2.

For this network, PDP works in a similar manner, except some of the U sets become smaller than DP. Again, node 2 initiates a broadcast. It's forward list is going to be exactly as previous. After that, node 3 re-broadcasts and it determines its P_3, U_3 and B_3 list as follows:

$$P_3 = N(N(3) \cap N(2)) = \emptyset \tag{3.16}$$

$$U_3 = N(N(3)) - N(3) - N(2) - P3 = \{5, 12\} \tag{3.17}$$

Table 3.3: The PDP Algorithm

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

$$B_3 = N(3) - N(2) = \{4, 11\} \tag{3.18}$$

Thus, the forward list becomes:

$$F(3) = \{4, 11\} \tag{3.19}$$

From Equation 3.12 and 3.19 we can see that the forward list of node 3 is exactly like the forwarding list of node 3 in DP. Then node 8 re-broadcasts the packet and P_8, U_8 and B_8 is constructed as follows:

$$P_8 = N(N(8) \cap N(2)) = \{2, 7, 8\} \tag{3.20}$$

$$U_8 = N(N(8)) - N(8) - N(2) - P_8 = \emptyset \tag{3.21}$$

$$B_8 = N(8) - N(2) = \{7, 9\} \tag{3.22}$$

Thus,

$$F(8) = \emptyset \tag{3.23}$$

Though there are some nodes in P_8 set, it does not affect the forwarding decision. For this sample network, the result of PDP is shown in Table 3.3, which gives the exact same output as DP. From Table 3.3, we can see that when node 4 and 11 forwards, the size of U_4 and U_7 in case PDP is smaller than the size of these U sets in DP.

Now, let us explain TDP algorithm for the same sample network of Figure 3.6. As node 2 is the source, the forward list of node 2 remains the same, which is $F(2) = \{3, 8\}$. When

Table 3.4: The TDP Algorithm

u	v	U_v	B_v	F(v)
$\emptyset$	2	4,7,9,11	1,3,8,10	3,8
2	3	5,12	4,11	4,11
2	8	$\emptyset$	7,9	[ ]
3	4	6	5	5
3	11	6,9	10,12	10,12
4	5	$\emptyset$	6	[ ]
11	10	1,8	2,9	2
11	12	$\emptyset$	6	[ ]

node 3 and 8 re-broadcasts, it gives the same output as DP and PDP. The details of number of forwarding for TDP are given in the Table 3.4.

From Table 3.4, we can see when node 5 receives the packet from node 4, it is in the forward list of node 4, $F(4) = \{5\}$. Note that, when node 5 forwards the packet, it gets:

$$U_5 = N(N(5)) - N(N(4)) = \emptyset \tag{3.24}$$

Thus, its forward list, $F(5)$ becomes empty. That means, node 5 only re-broadcasts for its neighbors but does not ask anyone to forward. The total number of forwarding for the sample network of Figure 3.6 using TDP algorithm is 8 (including the source broadcast), which is less than DP and PDP.

Now, let us trace the behavior of the proposed expanded neighbor information based dominant pruning, ExDP algorithm which uses 3-hop neighbor information to reduce re-broadcasts for the same sample network in Figure 3.6. Again consider node 2 is the source node which initiates a broadcast and starts with forward list, $F(2) = 3, 8$. The total list for the proposed method is illustrated in Table 3.5.

From Table 3.5, it is clearly visible that, upon receiving the broadcast packet from node 2 and being in the forward list, node 3 re-broadcasts the packet and calculates,

$$Q_3 = \emptyset \tag{3.25}$$

$$U_3 = N(N(3)) - N(N(2)) - Q_3 = \{5, 12\} \tag{3.26}$$

Table 3.5: The New Approach

u	v	Q_v	U_v	B_v	F(v)
∅	2	∅	4,7,9,11	1,3,8,10	3,8
2	3	∅	5,12	4,11	4,11
2	8	∅	∅	7,9	[ ]
3	4	∅	6	5	5
3	11	2,3,4,5,6,10,11,12	9	10,12	10
4	5	∅	∅	6	[ ]
11	10	∅	1,8	2,9	2

$$B_3 = N(3) - N(2) = \{4, 11\} \tag{3.27}$$

Q_3 set is empty because the other node in the forward list of node 2 is node 8 and which is not a neighbor of node 3. Thus, the forward list remains same as DP,

$$F(3) = \{4, 11\} \tag{3.28}$$

Now, when node 8 forwards,

$$Q_8 = \emptyset \tag{3.29}$$

$$U_8 = N(N(8)) - N(N(2)) - Q_8 = \emptyset \tag{3.30}$$

$$B_8 = N(8) - N(2) = \{7, 9\} \tag{3.31}$$

$$F(8) = \emptyset \tag{3.32}$$

The difference is seen when node 11 re-broadcasts the packet receiving from node 3. Forward list of node 3 is, $F(3) = \{4, 11\}$. Now, when node 11 forwards, it knows its neighbor 4 is also in the forward list of node 3 and $ID(4) < ID(11)$, so 2-hops away neighbors of node 4 is deducted from its U set. thus, it determines Q_{11}, U_{11} and B_{11} as follows:

$$Q_{11} = N(N(4)) = \{2, 3, 4, 5, 6, 10, 11, 12\} \tag{3.33}$$

$$U_{11} = \{9\} \tag{3.34}$$

$$B_{11} = \{10, 12\} \tag{3.35}$$

U_{11} set is reduced than DP, PDP and TDP. Thus node 11 asks only node 10 to forward to cover node 9 whereas in case of DP, PDP and TDP both nodes 10 and 12 were in the forward list of node 11. That's why, node 12 never forwards in case of ExDP. Finally it is clearly seen that total 7 broadcasts are required to convey the packet to all the nodes in the network using 3-hop neighbor connectivity information which is a visible improvement over DP, PDP and TDP.

Chapter 4

Experiments

In this chapter we present experimental results.

4.1 Performance Metrics

The example in the previous section shows that there are some cases when the proposed method, ExDP provides better result than DP, PDP and TDP. Now, to show how much improvement can be achieved some simulation experiments were carried out in random networks. The following metrics are used to evaluate the performance:

i. **Number of Transmissions Required:** Total number of transmissions required to reach the packet to all the nodes in the network. All the packets of the same broadcast assumed to have same user data and considered as same packet. Here, lower is better; the less the number of total transmissions, the more effective the method is.

ii. **Saved Rebroadcast (SRB):** SRB is represented as $r - t/r$; here r is the total number of nodes receiving the packet and t is the total number of nodes transmitting the packet.

iii. **Energy Consumption:** In a wireless ad hoc network the hosts are power constrained devices with limited energy and their power should be used in an efficient way. Thus, energy consumption is crucial in the design of new ad hoc routing protocols. If energy is not used in an efficient and effective way, any time the hosts may go off which may lead to network partitioning. Thus the goal should be reducing energy while broadcasting packets by reducing the number of broadcasting.

Undoubtedly, DP, PDP and TDP reduce total energy consumption by reducing re-broadcasting a packet than blind flooding where each node in the network broadcasts the packet exactly once. As the proposed optimized dominant pruning reduces transmissions than DP, PDP and TDP, it is expected that it also saves energy of the network as well as of the individual hosts.

Energy Required for Single Round of Hello Packets:

For ExDP, each node should have complete knowledge about its 3-hop neighbor connectivity information. These can be acquired by sending a hello packet with TTL field set to 3. Thus, when a node u sends a hello packet (with $TTL = 3$), all its neighbors, $N(u)$ receives the packet. Then these neighbors again forward the packet and all of the nodes in $N(N(u))$ receive the packet. Now, the nodes that are exactly 2-hop away from node u, $N(N(u)) - N(u)$ will forward the packet further and their 1-hop away neighbors which are actually 3-hop neighbors of u, $N(N(N(u))$ $[N(N(N(u)) - N(u))]$ will receive the packet. Thus, for ExDP, total number of hello packet transmission for single round, T and total number of hello packets reception for single round, R can be stated as follows:

$$T = \sum_{u=1}^{n} \left(1 + \left|N(u)\right| + \left|[N(N(u)) - N(u)]\right|\right) = \sum_{u=1}^{n} \left(1 + \left|NN((u))\right|\right) \tag{4.1}$$

And,

$$R = \sum_{u=1}^{n} \left(\left|N(u)\right| + \left|N(N(u))\right| + \left|N(N(N(u)) - N(u))\right|\right) \tag{4.2}$$

where n is the total number of nodes in the network.

For DP, the hello packets are propagated up to 2-hop. For DP total number of hello packet transmission for single round, T and total number of hello packets reception for single round, R will be calculated as,

$$T = \sum_{u=1}^{n} \left(1 + \left|N(u)\right|\right) \tag{4.3}$$

$$R = \sum_{u=1}^{n} \left(\left|N(u)\right| + \left|N(N(u))\right|\right) \tag{4.4}$$

As the hello packets are propagated up to 3-hop instead of 2-hop, there will be a little increase in energy consumption for hello packets propagation in case of ExDP proposed in this thesis than DP. However, it will not create a major issue as the hello packets are relatively very small than the broadcasting packets. Sometimes, it is required to lose a little to gain something more. It is shown next that, the extra energy required for hello packets will not affect the network as the total energy for broadcasting a data packet is reduced to a great extent than DP. Thus, by losing a little energy at the beginning a lot more energy is saved later by reducing the number of rebroadcasting a packet.

Total Energy Consumption for Broadcasting a Data Packet:

Total energy of a network is calculated as the summation of the total energy to transmit a packet in the network and the total energy for receiving the packet. Total energy can be stated as:

$$E_{\text{Total}} = E_T + E_R \tag{4.5}$$

where E_{T} is energy required for transmission and E_{R} is energy required for reception.

Now, energy required for the transmission in a network is defined by the energy consumed by all the transmitting nodes of the network. The neighbors of the forwarding nodes will receive the packet and for this energy required for reception is defined by the energy consumed by all these nodes while receiving the packet. Suppose, total i nodes will forward among all the nodes in the network and the forward list is, $F = \{F_1, F_{\,2}, F_3, ...F_{\text{i}}\}$.

Thus, if P_{t} and P_{r} is power required to transmit a packet by a node and power required to receive a packet by a node respectively, E_{T} and E_{R} of the network is defined as follows:

$$E_T = \Big|F\Big| \times P_t \times t \tag{4.6}$$

$$E_R = \sum_{F_i \in F} \Big|N(F_i)\Big| \times P_r \times t \tag{4.7}$$

Transmission power, P_t of a node depends on the transmission range of the node. The larger the transmission range or the distance the packet needs to be forwarded, the more power is required. So, $P_t = k.d\hat{n}$ for some appropriate constant k, where $n \geq 2$ is the path-loss exponent of outdoor radio propagation models and d is the distance between receiver and transmitter. For our experiment, we have chosen path-loss exponent $n = 4$,

Table 4.1: Parameters Used in Experiment

Parameter	Value
Number of nodes	100-500
Area	650m $\times 650m$
Transmission range	125m-225m
Transmission rate	2Mbps
Broadcast data packet size	1000 bytes
Hello packet size	20 bytes
Path loss exponent, n	4
Transmission power of a node (1m)	-70 dBm (0.0000001mW)
Receiver power	80 mW

minimum transmit power required for 1m is $-70dBm(0.0000001mW)$ and fixed receiver power $80mW$ is consumed at each node.

4.2 Experimental Environment

To evaluate the performance, we deployed random networks consisting of 100-500 nodes distributed over $650m \times 650m$ square region. The transmission range was limited from 125m to 225m for each network. An ideal MAC layer was assumed, i.e., contention and collisions were not modelled. The transmission rate was 2Mbps and the size of all data packets was same which was equal to 1000 bytes. The size of hello packets was assumed to be 20 bytes.

For test, each node was placed randomly in the simulation space. The nodes remained in their position for the whole time. Each time, node 1 initiated a broadcast packet. Any random node could have been selected to initiate the broadcast, but this does not matter as the position of node 1 was randomly placed each time. Total ten different scenarios were generated for each case. For example, ten random networks were generated for 100 nodes with 125m transmission range (same transmission range for all nodes) The average value of these ten results is shown on the graph to carry out the comparison between the schemes. The simulation code was built using C++ programming language.

The networks were always generated as connected network. Depending on the number of nodes the networks can be categorized as sparse network, moderately dense network and dense network.

The parameters and their values used in the experiment are summarized in Table 4.1.

4.3 Experimental Result regarding Reducing Broadcast

4.3.1 Number of Transmissions Required

i. **Effect of Transmission Range:**

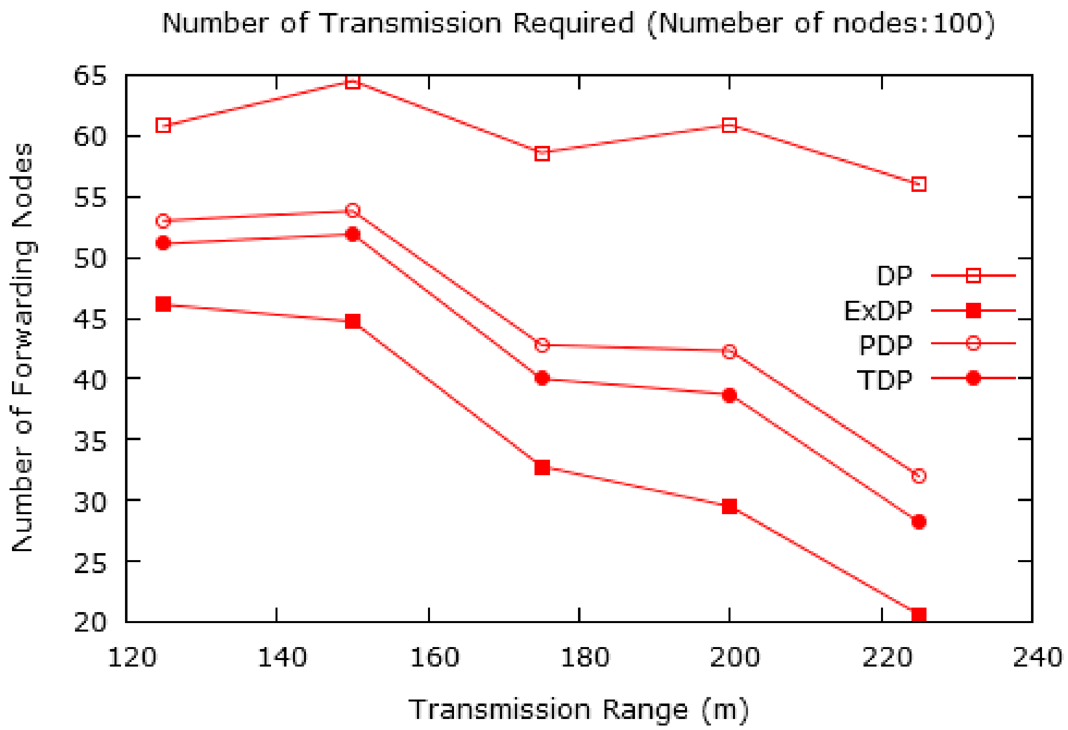

Figure 4.1: Performance comparison of DP, PDP, TDP and ExDP in case of number of forwarding nodes for sparse network

From Figure 4.1, we can see the effect of different methods in a sparse network. It shows the result for 100 nodes having transmission range between 125m to 225m. Using dominant pruning 50-65 broadcasts are required on an average to forward the message to the entire network; whereas 20-45 forwarding is enough while using ExDP. TDP and PDP provides very close result but both require more forwarding than ExDP. It is clearly visible that DP shows the worse performance among the four methods. The proposed algorithm, ExDP generates best results in all transmission ranges to complete a broadcast. As the transmission range increases, mainly the number of forwarding decreases for all methods. It is as expected because with the increase in transmission range, a node can cover more neighbors with a single forwarding than using smaller transmission range.

Figure 4.2 shows the effect of all the methods for 300 nodes deployed in $625m \times 625m$

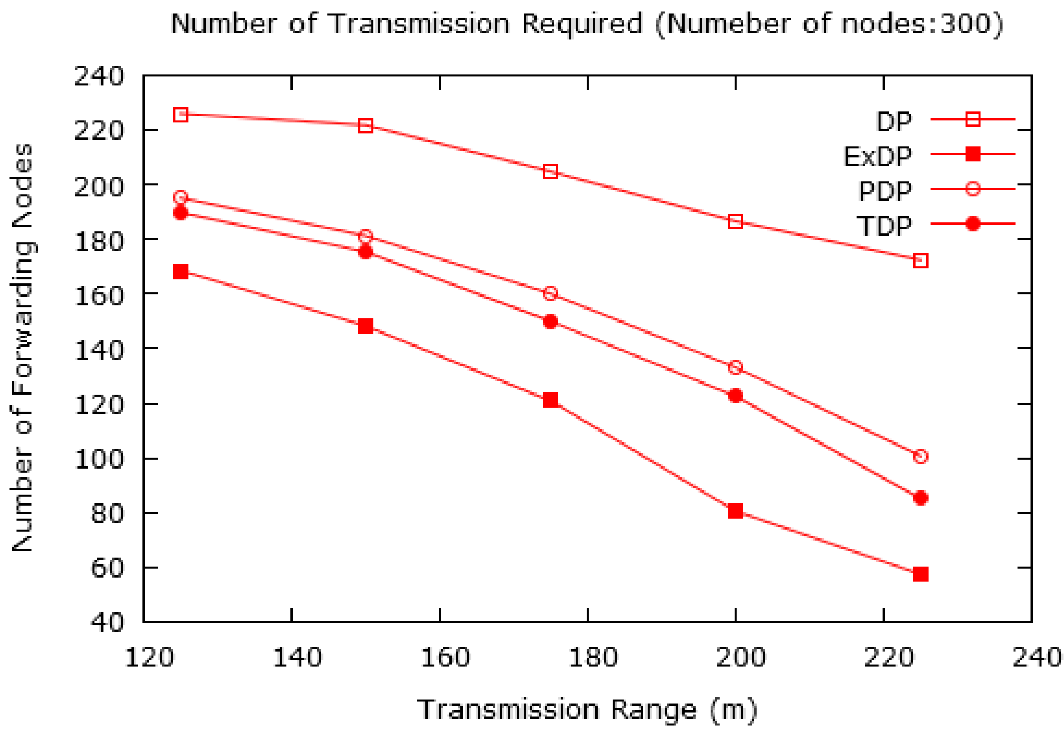

Figure 4.2: Performance comparison of DP, PDP, TDP and ExDP in case of number of forwarding nodes for moderately dense network

region. Again, 10 scenarios were generated for each transmission range and the average value is taken to plot on the graph. It is evident that, with the increasing transmission range, the performance of ExDP gets better than others. With larger transmission range, a node has more neighbors in its transmission range than a node with smaller transmission range. Thus, it is obvious that the number of neighbors of a node already covered by other nodes increases. Thus, using 3-hop neighbor information ExDP can detect those redundant transmissions more than DP, PDP and TDP and do not include those covered nodes in the uncovered set. When transmission range is 225m, DP requires almost 172 nodes to forward the message whereas ExDP requires only 57 nodes to forward. So, ExDP saves almost 115 rebroadcasts in this case.

Now, let's see the effect of the methods on dense networks. Dense networks were generated by adding more nodes to the network while keeping the other settings similar. Figure 4.3 presents the effect of the schemes on a dense network with 500 nodes within the same deployment area of $625m \times 625m$. Here transmission ranges are similar from 125m to 225m like earlier and again average of 10 scenarios are considered. In the dense network of 500 nodes, ExDP outperformed other three methods in a similar manner. It performed even

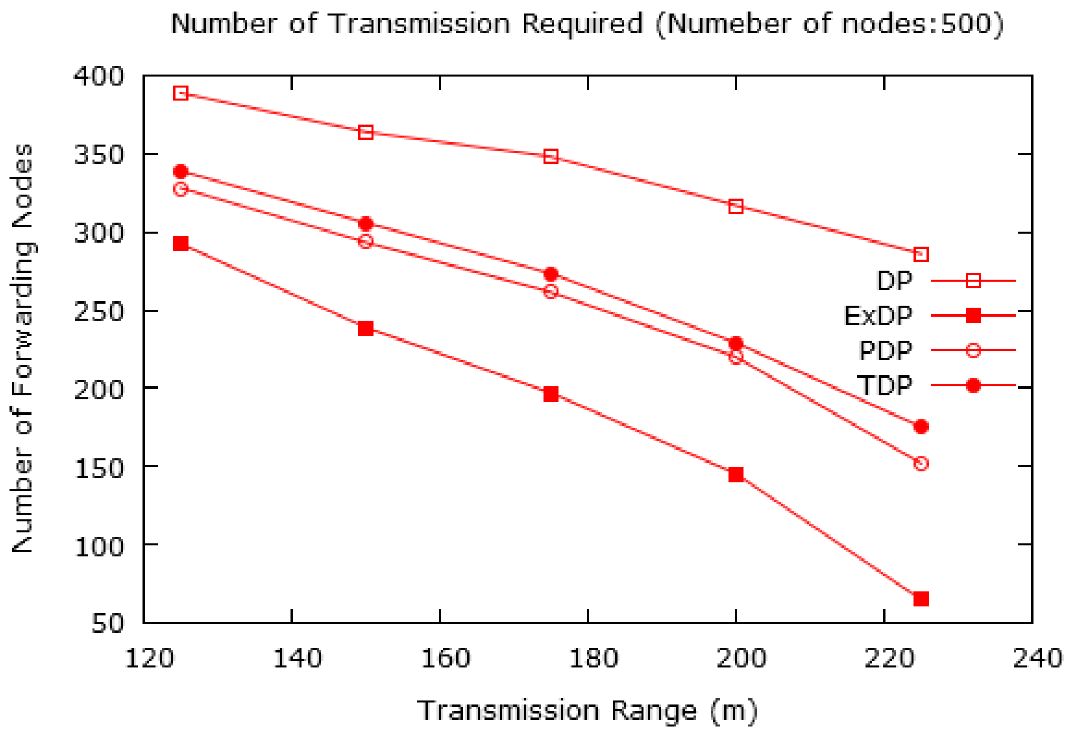

Figure 4.3: Performance comparison of DP, PDP, TDP and ExDP in case of number of forwarding nodes for dense network

better with increasing transmission range. At 125m transmission range DP asks almost 388 nodes to forward the message. PDP and TDP need 338 and 328 nodes to rebroadcast respectively. However, ExDP requires almost 292 nodes to forward among 500 nodes. With increasing transmission range at 225m, in case of DP, PDP and TDP 286, 175 and 152 nodes forward the same packet to cover all the nodes in the network respectively. Whereas, ExDP outperforms all the three methods by asking only 65 nodes to forward among the 500 nodes. Thus, we can see the differences among DP, PDP, TDP and ExDP in sparse network, moderately dense network and dense network with the similar settings. The optimized one performs much better than other three in all the cases.

ii. **Effect of Node Density:** Figure 4.4 shows the effect of node density on all four methods. In this case, the transmission range is kept fixed to 175m for all nodes but each scenario has different number of nodes ranging from 100 to 500. It is clearly evident as the number of nodes increases in a network, number of forwarding nodes also increases linearly for all the methods. Though ExDP also gives linear increase with the increasing number of nodes, its growth is less than DP and other two algorithms. Thus, in case of node density ExDP gives better result than DP, PDP and TDP.

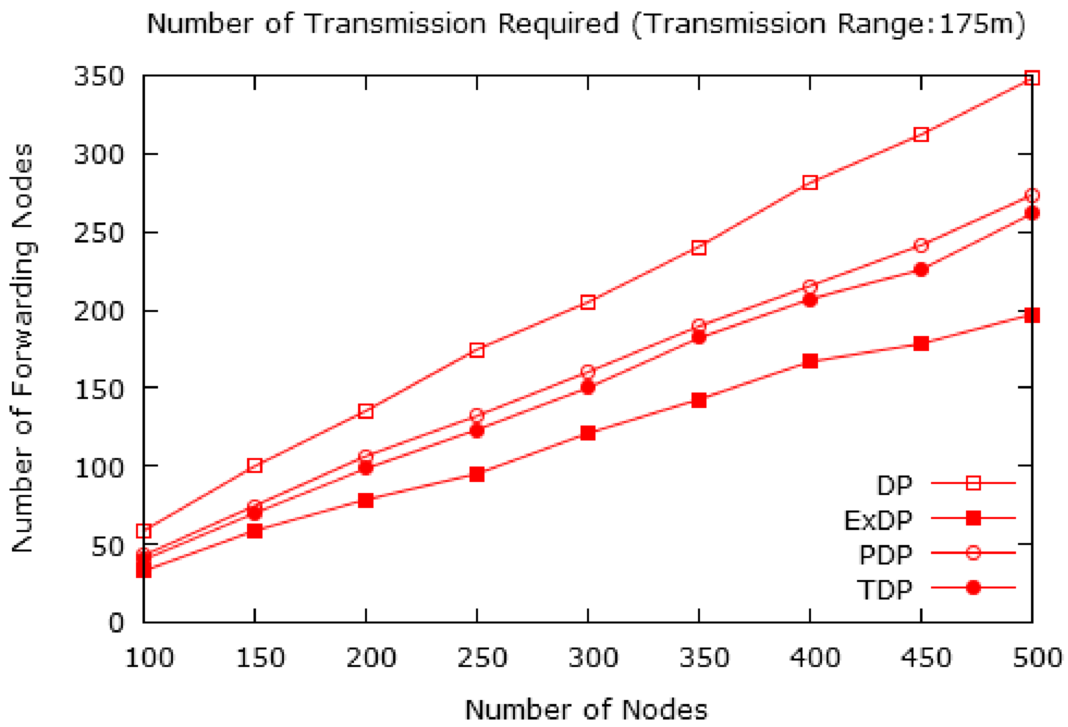

Figure 4.4: Performance comparison of DP, PDP, TDP and ExDP in case of node density

The same scenario is shown in Figure 4.5 having transmission range set to 225m. Here, with the increasing amount of nodes in the network, the efficiency of DP decreases. PDP and TDP perform better than DP. On the other hand, ExDP outperforms the three algorithms. Although the number of forwarding nodes increases linearly with the increasing number of nodes in case of DP, PDP and TDP, the growth of ExDP is almost negligible in case 225m transmission range.

In summary, the proposed method, ExDP gives better performance than DP, PDP and TDP by reducing number of forwarding nodes with increase in network parameters such as transmission range and network density.

4.3.2 Saved Rebroadcast (SRB)

i. **Effect of Transmission Range:**

Figure 4.6 shows how much rebroadcasts are saved for 100 nodes. This gives an idea how much efficient a scheme is when compared to blind flooding. In case of blind flooding, whenever a node receives a packet for the first time, it rebroadcasts it without any further

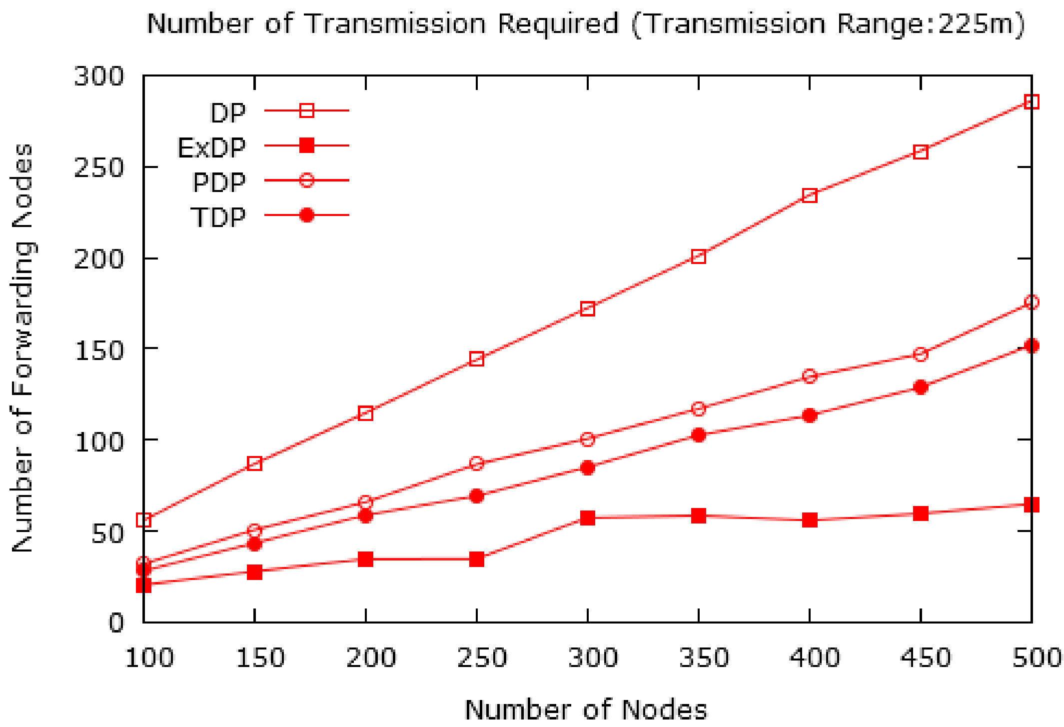

Figure 4.5: Performance comparison of DP, PDP, TDP and ExDP in case of node density

consideration. Thus, if the result of blind flooding was plotted in this graph, it would always have value of 0.

In Figure 4.6, SRB is plotted for DP, PDP, TDP and ExDP for 100 nodes having transmission range from 125m to 225m. Similarly, 10 different scenarios have been taken for each transmission range and node 1 initiated the broadcast every time. The mean value of these 10 scenarios is plotted in the graph.

PDP saves from 8% rebroadcasts (125m transmission range) to 24% (225m transmission range) than DP and TDP is able to save from 10% (125m) to 28%(225m) compared to DP. Now, ExDP is able to reduce the number of broadcast required in network of 100 nodes from 15% (125m) to 35%(225m) compared to DP which is a clear improvement over PDP and TDP.

Figure 4.7 represents the number of saved rebroadcast for a moderately dense network with 300 nodes. Similarly, in this case with the increasing number of transmission range the percentage of SRB increases. For the network with 300 nodes, ExDP proposed in this thesis is able to save from 19% (125m) up to 39% (225m) than dominant pruning algorithm.

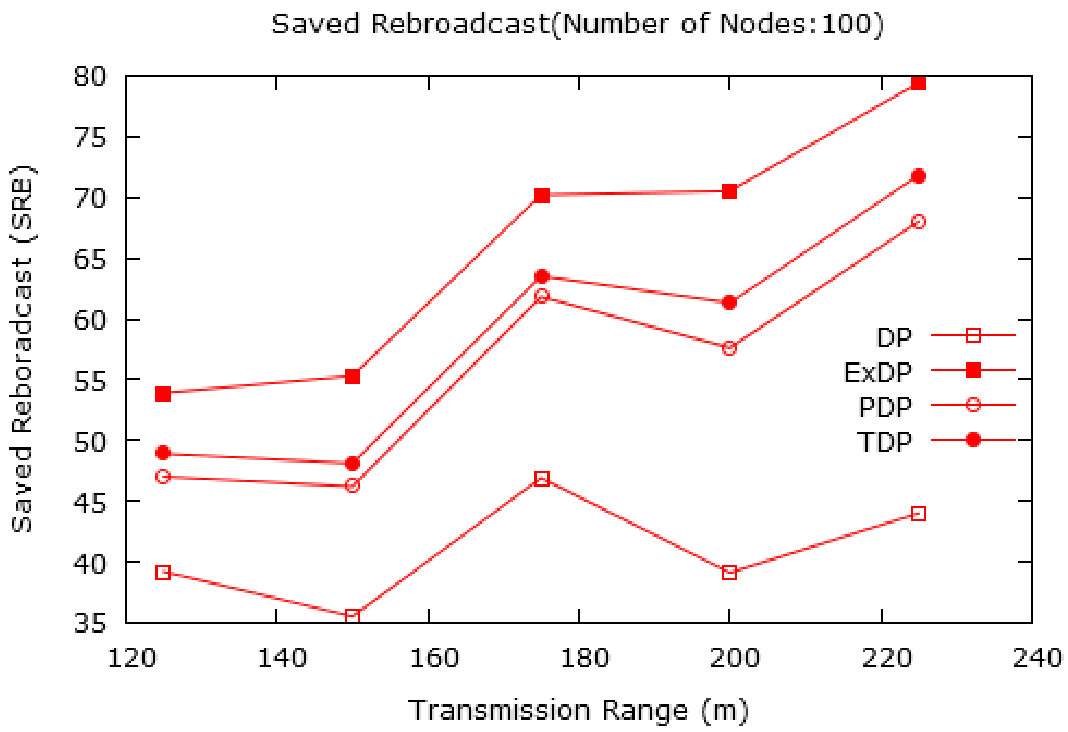

Figure 4.6: Performance for SRB of DP, PDP, TDP and ExDP in case of transmission range for sparse network

Figure 4.8 shows the amount of saved rebroadcast for a dense network. Networks having 500 nodes also improve efficiency by saving rebroadcasts from 20% in case of lower transmission range to 44% in case of larger transmission range. Here, only the number of nodes is greater than other networks, otherwise all the scenarios are same as before.

From all the cases above, it can be summarized that all the methods save rebroadcast than blind flooding with the increase in transmission range for all kinds of networks- sparse, moderately dense or dense network. The larger the transmission range, the more rebroadcast is saved.

ii. **Effect of Node Density:**

Figure 4.9 represents the effect of node density about how many rebroadcasts are saved for the four different methods. In this case, the transmission range is fixed to 175m and nodes vary from 100 to 500. It is clear from Figure 4.9 that, all the methods save rebroadcasts than bind flooding but ExDP exceedingly outperforms the other three algorithms.

Figure 4.10 shows the effect of node density in case of SRB having transmission range fixed to 225m. Dominant pruning saves from 41% to 44% broadcasting than blind flooding

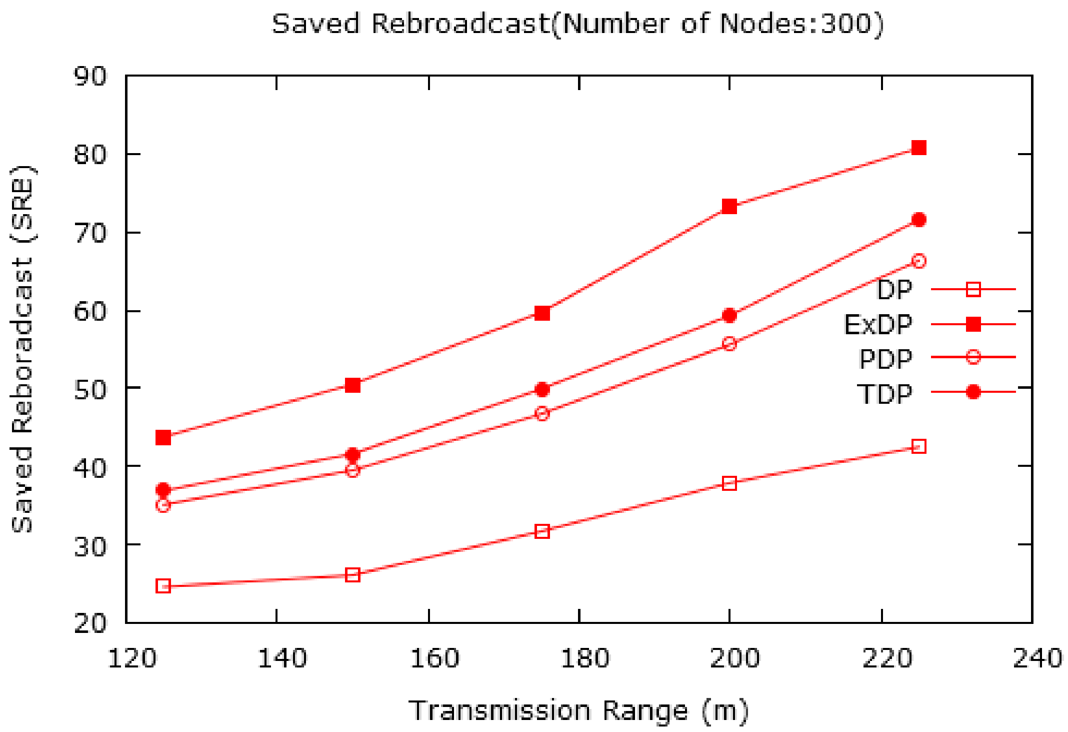

Figure 4.7: Performance for SRB of DP, PDP, TDP and ExDP in case of transmission range for moderately dense network

for networks having nodes 100 to 500. On the other hand, ExDP saves almost from 79% (for networks with 100 nodes) to 88% (for networks having 500 nodes) than blind flooding. Actually, the number of nodes does not affect much in case of saving broadcasts.

4.4 Experimental Result Regarding Energy Consumption

4.4.1 Broadcasting Hello Packets

i. **Effect of Transmission Range:**

Figure 4.11 shows energy required for propagating hello packets for a single round with networks having 100 nodes for DP, PDP, TDP and ExDP. Here, we can see that ExDP requires a little more energy than DP for propagating hello messages up to 3-hop. It requires additional 0.375 mJ (3.342 mJ for DP, 3.717 mJ for proposed) when transmission range is set to 125m for all nodes. And it keeps growing with the transmission range. When transmission range is set to 225m it needs additional 10.452 mJ for single round of hello packet having TTL field set to 3.

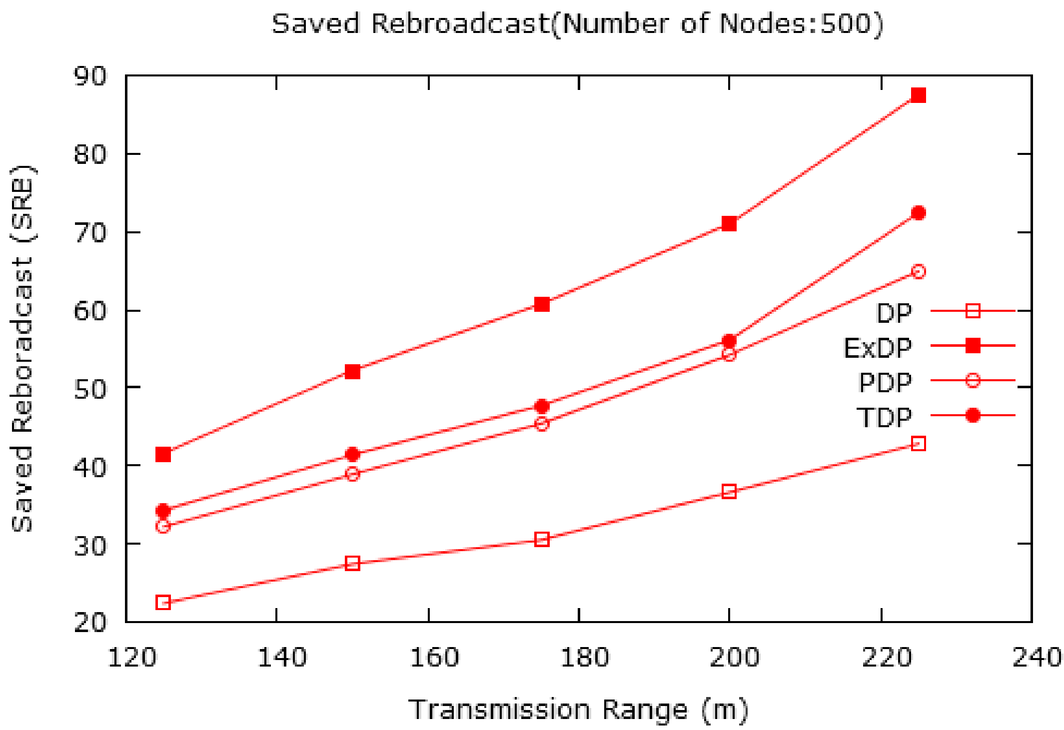

Figure 4.8: Performance for SRB of DP, PDP, TDP and ExDP in case of transmission range for dense network

Figure 4.12 shows, the energy required for a single round of hello packets for a moderately dense network having total 300 nodes.

From Figure 4.13, we can see the additional energy required for ExDP than DP for dense networks. In this case, similarly ExDP requires more energy than DP and comparing it with Figure 4.18, it is visible that additional energy for hello packet will not affect the total performance of the network as total energy for broadcasting a data packet in ExDP is reduced much than DP.

As the proposed method needs to propagate hello packets to 3-hop instead of 2-hop like DP, it consumes a little more energy than DP. Still because of the size of hello packets, this little extra consumed energy will not affect the overall performance of the proposed algorithm.

ii. **Effect of Node Density:**

Figure 4.14 and 4.15 shows the effect of energy consumed for propagating hello packets in case of node density. Here, transmission range is fixed to 175m and 225m respectively. Nodes vary from 100 to 500. With the increasing number of nodes, the consumed energy

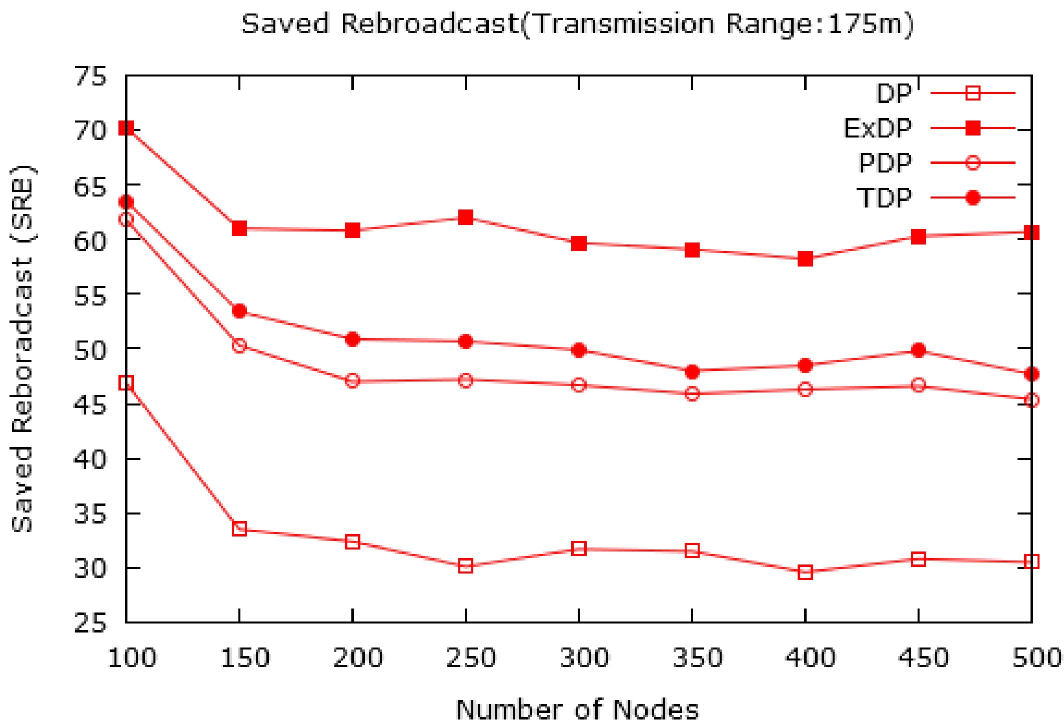

Figure 4.9: Performance for SRB of DP, PDP, TDP and ExDP in case of node density

for propagating a single round of hello packets keeps growing.

Thus, it is clearly evident that, ExDP requires a little extra overhead for collecting 3-hop neighbor information by sending hello packets up to 3-hops than DP, PDP and TDP. However, in the next section, it is shown that, the total consumed energy is very less for ExDP than others while sending a broadcast data packet in return of this extra overhead.

4.4.2 Broadcasting Data Packets

i. **Effect of Transmission Range:**

Figure 4.16 shows the total energy consumed for broadcasting a packet for a sparse network having 100 nodes. The transmission range is same as earlier from 125m to 225m. As stated previously, 10 networks were generated with 100 nodes for each transmission range and value was taken for all scenarios; at the end the average of all 10 measures are taken. Here, with the increasing value of transmission range total consumed energy by DP also increases. On the contrary, total energy consumed by ExDP decreases as the transmission range increases. ExDP reduced re-broadcasts than other methods. With the reduction in

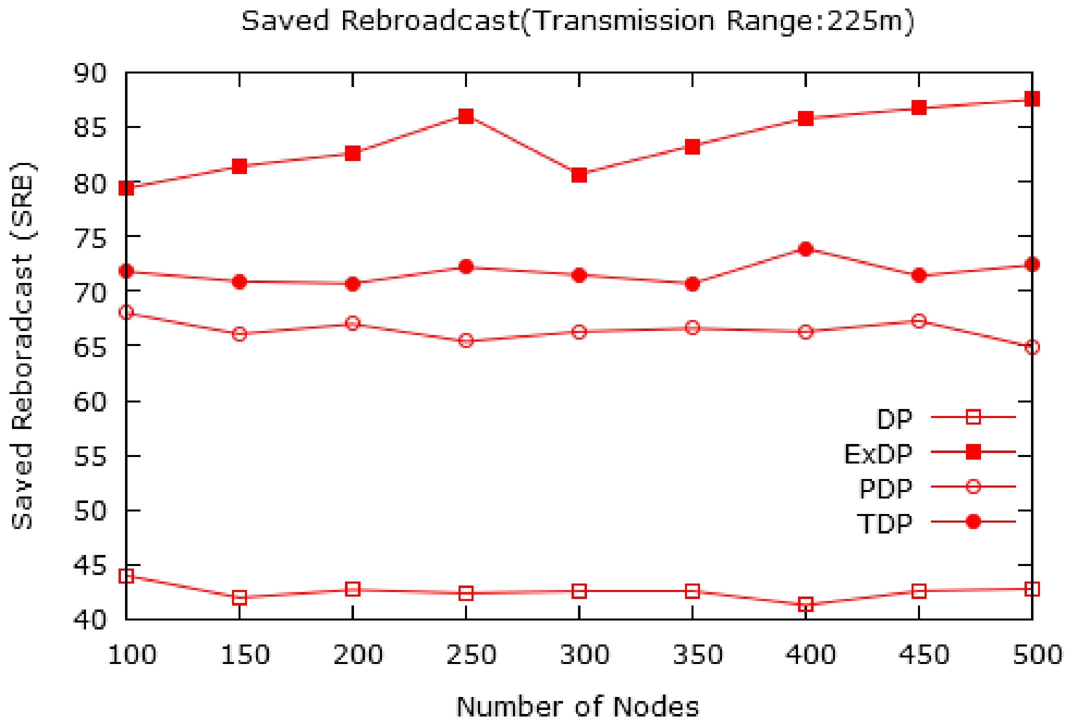

Figure 4.10: Performance for SRB of DP, PDP, TDP and ExDP in case of node density

transmitting, also the total number of reception reduces. Thus, it is easily understandable that, ExDP will reduce energy consumption of the whole network to a great extent.

Comparing Figure 4.16 with Figure 4.11, it is clearly visible that, this small additional increase in energy while sending hello packets is negligible to the amount of energy reduced while broadcasting data packets. For example, when transmission range is set to 225m almost 54 mJ is reduced in proposed method than DP; therefore, additional 10.5 mJ will not affect the efficiency of ExDP.

Figure 4.17 represents the total consumed energy for networks having 300 nodes. DP requires almost 311 mJ energy at lower transmission range (125m) which keeps growing with the transmission range and finally becomes 689 mJ at larger transmission area (225m). In case of PDP total energy required to broadcast a packet is from 276 mJ (at 125 m) to 434 mJ (at 225m), whereas TDP is able to reduce energy consumption than earlier two, from 270 mJ (125m) to 378 mJ (225m). However, ExDP performs exceedingly well than these three methods. As the transmission range increases the total energy consumed is decreased. Only 226 mJ energy is consumed in this case for 225 m transmission range which is a clear improvement than others.

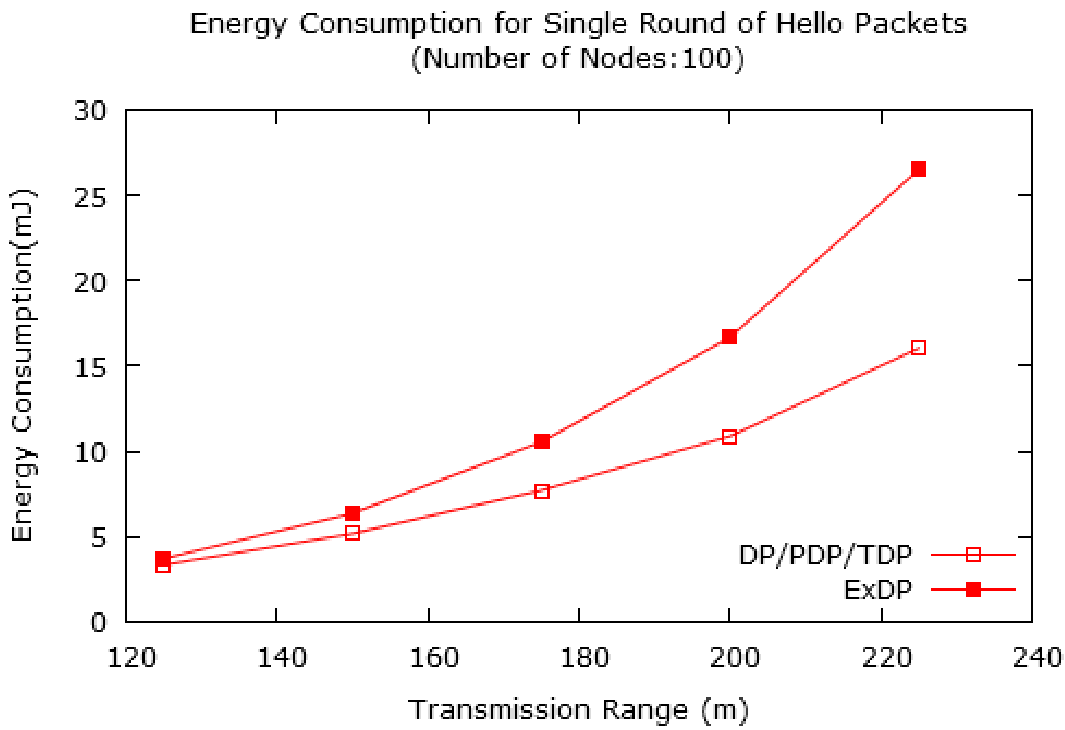

Figure 4.11: Energy consumption while propagating hello packets for single round for DP, PDP, TDP and ExDP for a sparse network

Figure 4.18 shows the total required energy of dense networks for broadcasting a packet such that the packet reaches to all the nodes of the network at least once. In the denser network having 500 nodes, ExDP outperforms other methods similarly.

ii. **Effect of Node Density:**

From Figure 4.19 we can see the behavior of total energy consumed for DP, PDP, TDP and ExDP in terms of node density. Here, transmission range is kept fixed to 175m for all networks. As the network becomes dense the total energy consumed increases exponentially. For DP, networks with 100 nodes require almost 56 mJ to broadcast a single packet to all the nodes and with 500 nodes 1469 mJ are needed. On the other hand, ExDP is able to reduce the total consumed energy from 56 mJ to 31 mJ for 100 nodes and from 1469 mJ to 900 mJ for networks with 500 nodes.

In Figure 4.20, the transmission range is fixed to 225m and other settings are same as similar. Similarly as previous case, ExDP reduces total energy than others as number of odes increases.

Thus, it can be summarized that, with the increasing number of nodes in a network, ExDP

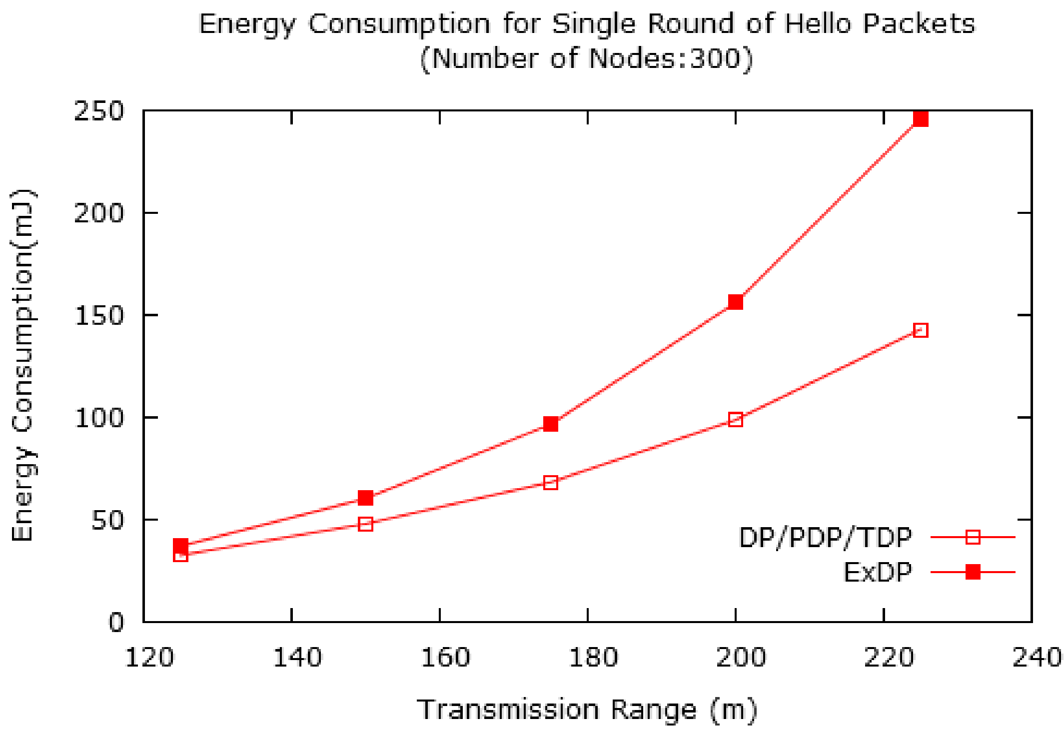

Figure 4.12: Energy consumption while propagating hello packets for single round for DP, PDP, TDP and ExDP for moderately dense network

requires less energy to broadcast a packet to all the nodes in the network. Also, if the transmission range in increased, the energy is reduced more.

In summary, it is clearly evident that using 3-hop neighbor connectivity information ExDP outperforms DP, PDP and TDP by reducing total consumed energy in case of both transmission range and node density.

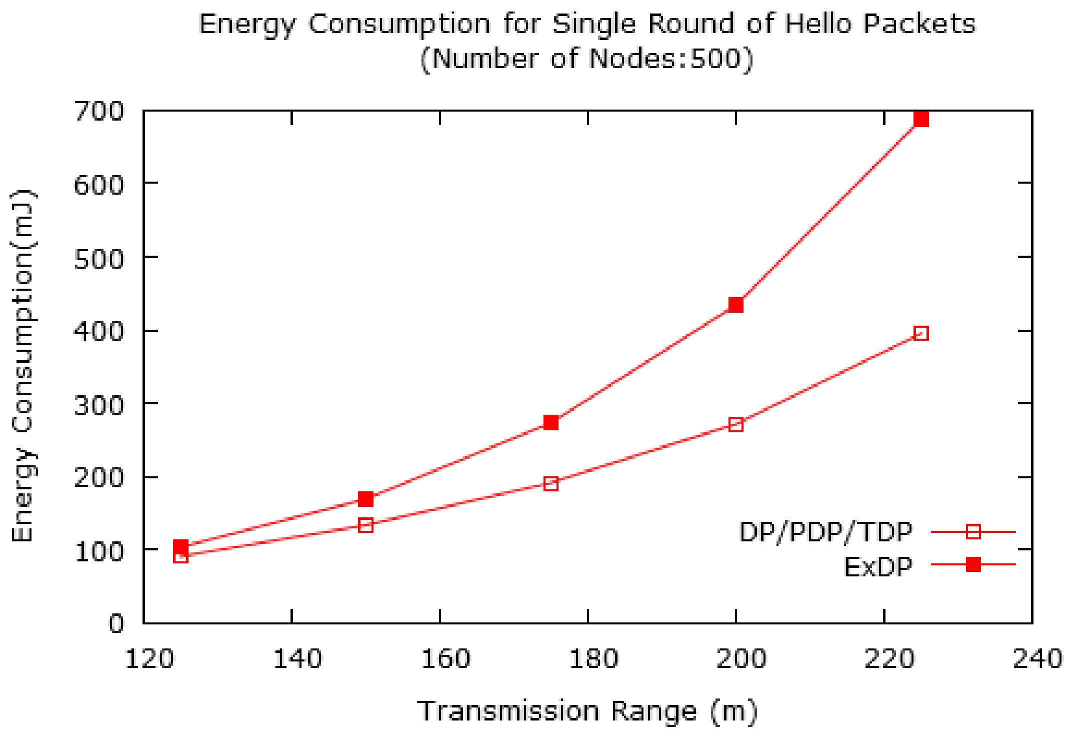

Figure 4.13: Energy consumption of propagating hello packets for single round for DP, PDP, TDP and ExDP for dense network

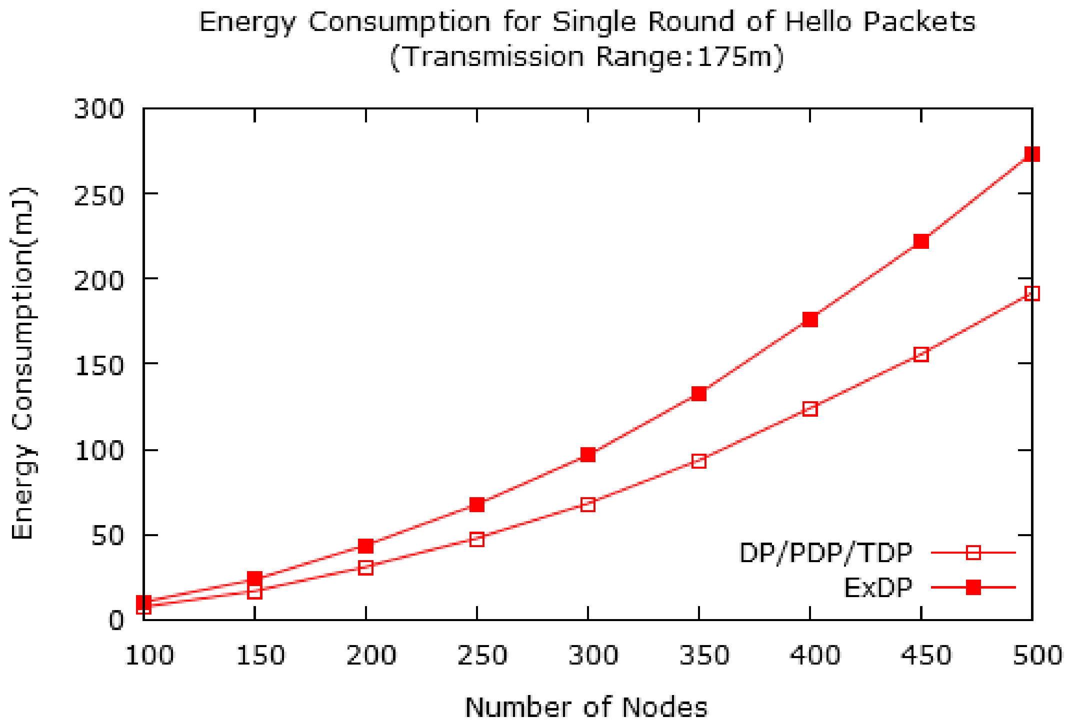

Figure 4.14: Energy consumption while propagating hello packets for single round for DP, PDP, TDP and ExDP varying number of nodes

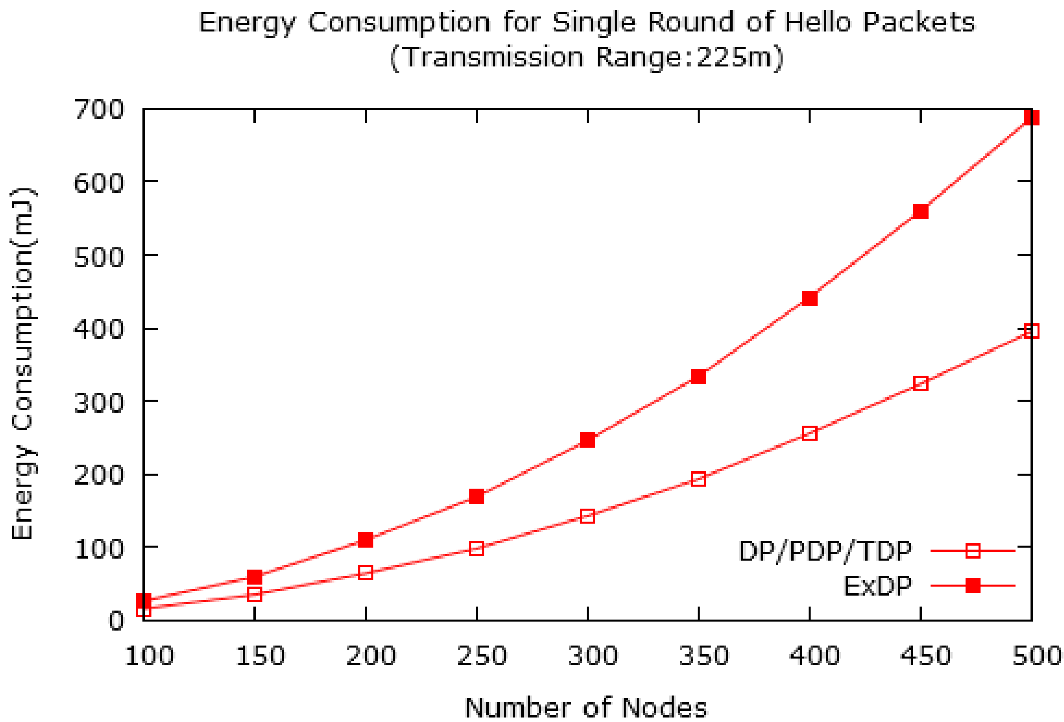

Figure 4.15: Energy consumption while propagating hello packets for single round for DP, PDP, TDP and ExDP varying number of nodes

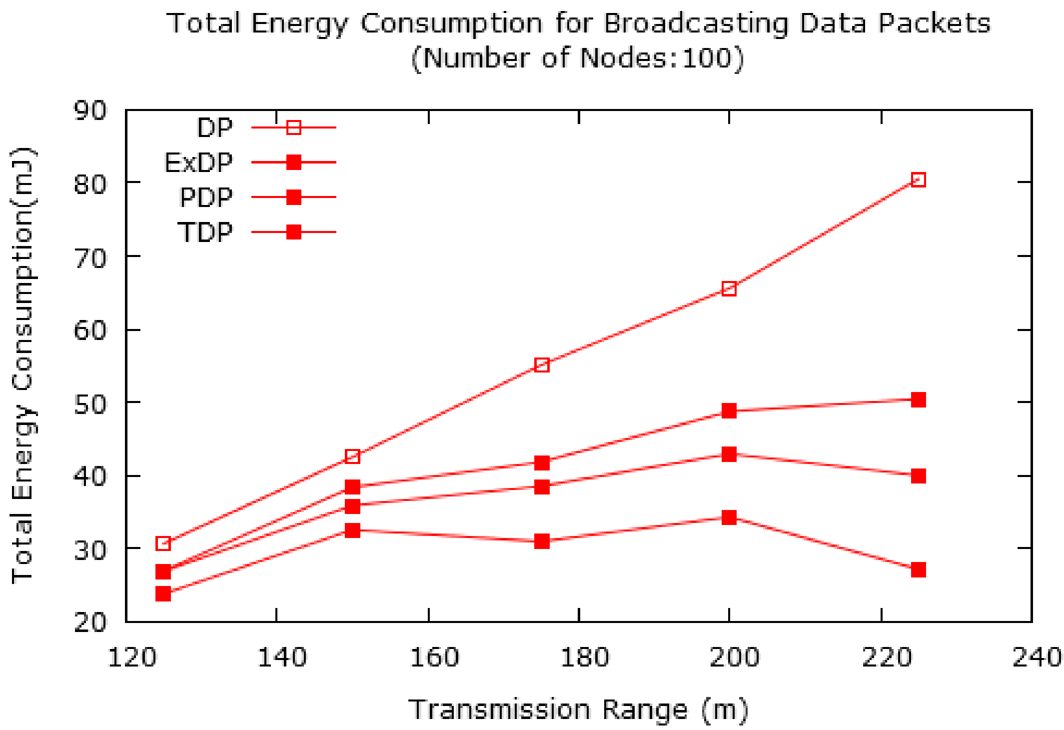

Figure 4.16: Total energy consumption while broadcasting a data packet for DP, PDP, TDP and ExDP in case of sparse network

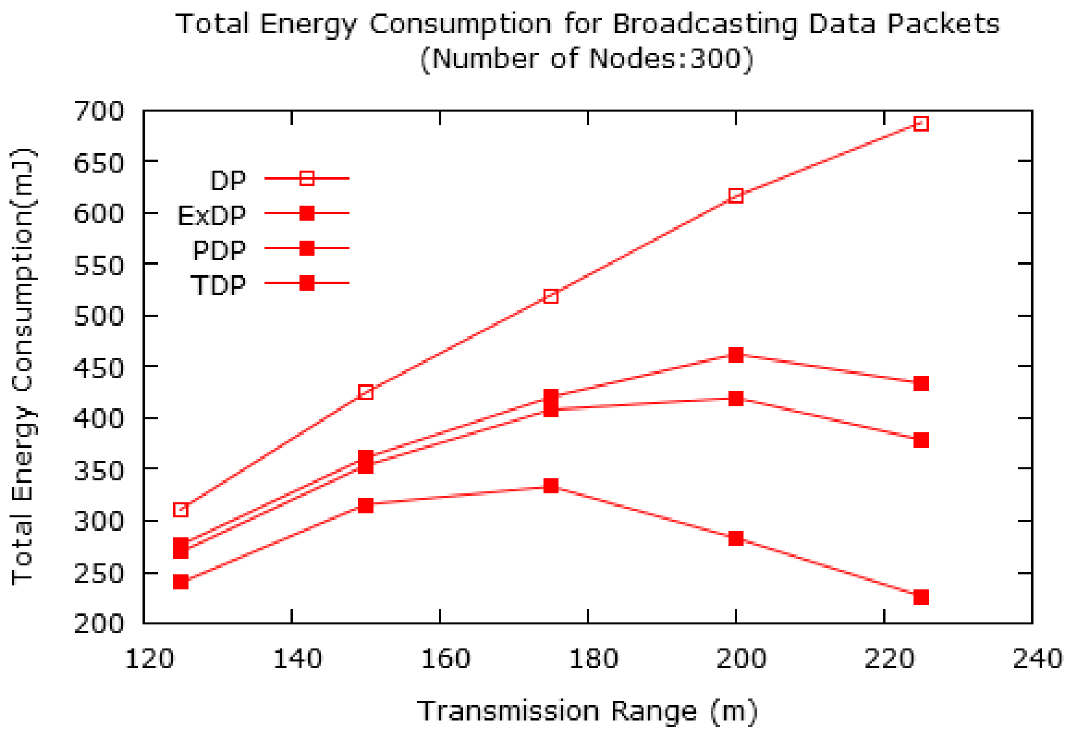

Figure 4.17: Total energy consumption while broadcasting a data packet for DP, PDP, TDP and ExDP in case of moderately dense network

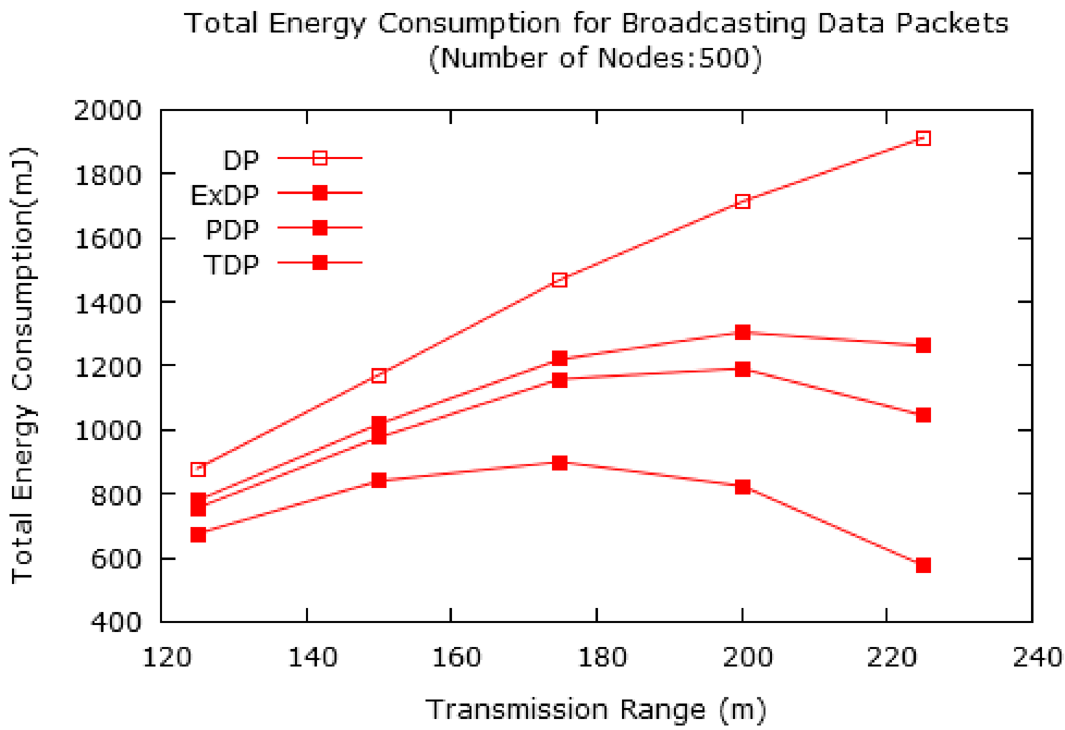

Figure 4.18: Total energy consumption while broadcasting a data packet for DP, PDP, TDP and ExDP in case of dense network

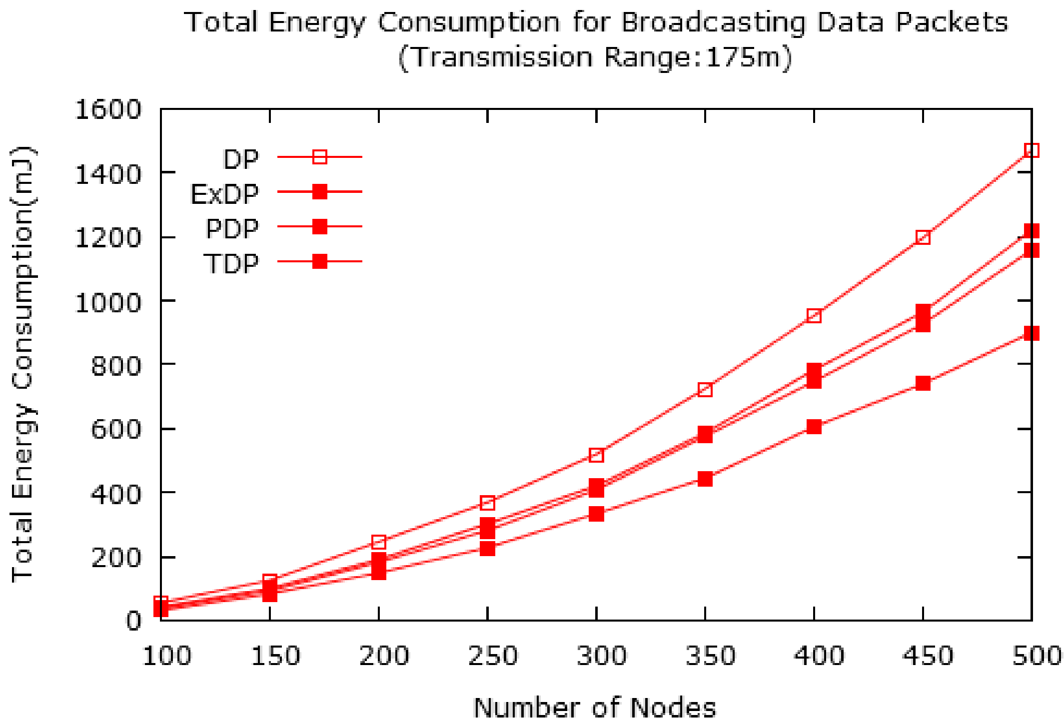

Figure 4.19: Total energy consumption while broadcasting a data packet for DP, PDP, TDP and ExDP varying number of nodes

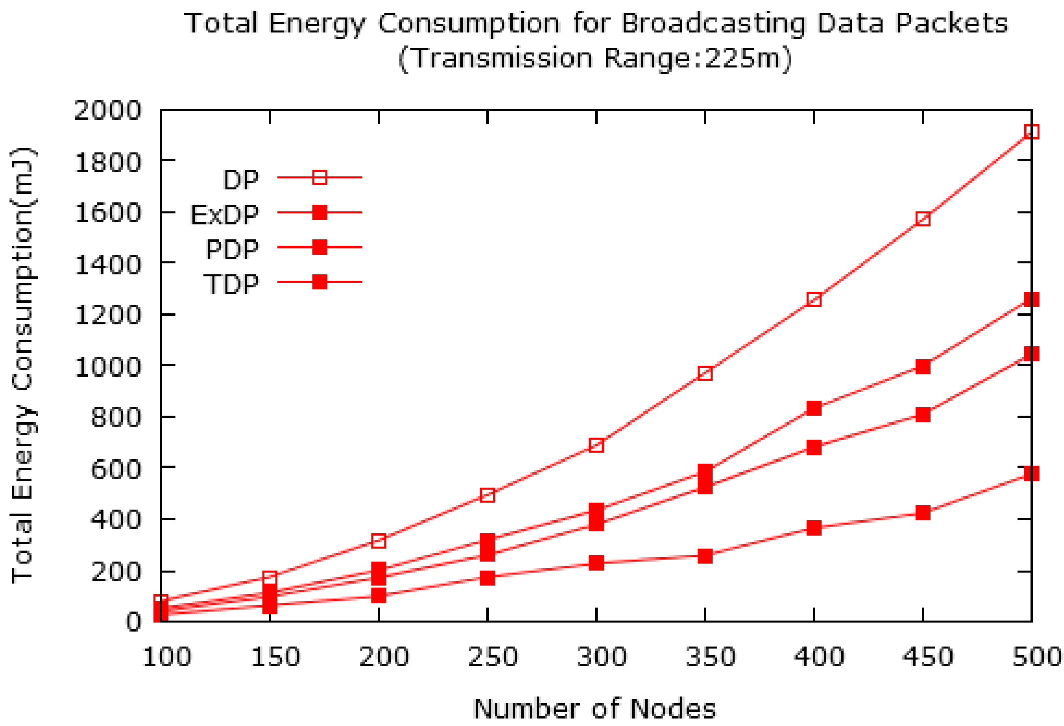

Figure 4.20: Total energy consumption while broadcasting a data packet for DP, PDP, TDP and ExDP varying number of nodes

Chapter 5

Conclusion and Future Work

In this thesis, an efficient broadcast method for wireless ad hoc networks called *Expanded neighbor information based dominant pruning, ExDP* is proposed to mitigate broadcast storm problem by reducing rebroadcasting in a network. This method provides full network coverage as long as the network is connected. The new method, ExDP improves an existiing broadcast technique, dominant pruning (DP) by expanding neighbor connectivity information of nodes in an ad hoc network. Both DP and ExDP require only some of the node of the network to forward the message, while other nodes simply receive it. Each time a node forwards a message, all of it's one hop neighbors receive it and the message header contains information about future forwarding. ExDP basically restores all the properties of DP except it uses 3-hop neighbor information of each node belongs to the network whereas DP uses 2-hop connectivity information. This is done by making the considered forwarding node v aware of other forwarding nodes' coverage of the previous hop when U_v set is determined.

Our simulation shows superiority of ExDP over previous algorithms DP and two heuristics-PDP and TDP in terms of saving rebroadcasts, number of required forwarding nodes and total energy consumed in a network while broadcasting a data packet. Although ExDP increases a little overhead while collecting 3-hop neighbor information, this overhead is insignificant by the energy savings resulting from fewer number of required re-broadcasts. All the four algorithms were carried out on networks which had 100 nodes to 500 nodes. In all the networks, ExDP was very effective in reducing broadcast redundancy while maintaining reach-ability the packet to all the nodes of the network. From the previous chapter, it was clearly evident that, other than a little overhead while broadcasting HELLO packets, ExDP outperforms DP, PDP and TDP. For the networks used in the simulation having 100-500 nodes, ExDP saves almost 80% broadcasts than blind flooding for a transmission range of 225m whereas DP saves only 40% than blind flooding. It is just a single scenario but from this, we can easily understand that, ExDP saves rebroadcast and consumes less energy than DP and blind flooding by reducing unnecessary rebroadcasts.

In future, we can extend the neighbor knowledge from 3-hop to k-hop. This thesis work only considered static networks while conducting simulations. Future research can be conducted in mobile scenarios to see the effect of the proposed method in a high-mobility network. The 3-hop neighbor information may become stale in mobile scenario, a way can be suggested in future to handle the stale information.

Bibliography

[1] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu, “The broadcast storm problem in a mobile ad hoc network,” *Wireless networks*, vol. 8, no. 2, pp. 153–167, 2002.

[2] H. Lim and C. Kim, “Flooding in wireless ad hoc networks,” *Computer Communications*, vol. 24, no. 3, pp. 353–363, 2001.

[3] S. Guha and S. Khuller, “Approximation algorithms for connected dominating sets,” *Algorithmica*, vol. 20, no. 4, pp. 374–387, 1998.

[4] H. Liu, P. Wan, X. Jia, X. Liu, and F. Yao, “Efficient flooding scheme based on 1-hop information in mobile ad hoc networks,” in *INFOCOM 2006. IEEE International Conference on Computer Communications Proceedings Barcelona,Spain April 23-29*, 2006.

[5] K. Utsu and H. Ishii, “Load-aware flooding over ad hoc networks enabling high message reachability and traffic reduction,” in *International conference on mobile computing and ubiquitous networking (ICMU) Seattle, U.S.A April 26-28*, 2010.

[6] I. Hur, T.-D. Le, M.-H. Jo, and H.-S. Choo, “Regular tiling based flooding scheme for energy efficiency in wireless ad hoc networks,” *Journal of Internet Technology*, vol. 13, no. 2, pp. 257–271, 2012.

[7] V. Paruchuri, A. Durresi, and R. Jain, “Optimized flooding protocol for ad hoc networks,” *arXiv preprint cs/0311013*, 2003.

[8] J. Arango, M. Degermark, A. Efrat, and S. Pink, “An efficient flooding algorithm for mobile ad-hoc networks,” in *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt) Cambridge, UK March 24-26*, 2004.

[9] Y. Yi, M. Gerla, and T.-J. Kwon, "Efficient flooding in ad hoc networks using on-demand (passive) cluster formation," *contract*, vol. 14, no. 01-C, p. 0016, 2002.

[10] T. J. Kwon and M. Gerla, "Efficient flooding with passive clustering (PC) in ad hoc networks," *ACM SIGCOMM Computer Communication Review*, vol. 32, no. 1, pp. 44–56, 2002.

[11] W. Lou and J. Wu, "On reducing broadcast redundancy in ad hoc wireless networks," in *Hawaii International Conference on System Sciences, (HICSS) Hawaii,U.S.A January 6-9*, 2003.

[12] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint relaying for flooding broadcast messages in mobile wireless networks," in *Hawaii International Conference on System Sciences, (HICSS) Hawaii,U.S.A January 7-10*, 2002.

[13] H. Lim and C. Kim, "Multicast tree construction and flooding in wireless ad hoc networks," in *International workshop on Modeling, analysis and simulation of wireless and mobile systems (MSWiM) Boston, MA, USA August 20-22*, 2000.

[14] M. A. Spohn and J. Garcia-Luna-Aceves, "Enhanced dominant pruning applied to the route discovery process of on-demand routing protocols," in *International Conference on Computer Communications and Networks (ICCCN) Texas, USA October 20-22*, 2003.

[15] A. Rahman, M. E. Hoque, F. Rahman, S. K. Kundu, and P. Gburzynski, "Enhanced partial dominant pruning (EPDP) based broadcasting in ad hoc wireless networks.," *Journal of Networks*, vol. 4, no. 9, pp. 895–904, 2009.

[16] G.-X. Kok, C.-O. Chow, and H. Ishii, "Reducing broadcast redundancy in wireless ad-hoc networks with implicit coordination among forwarding nodes," *Wireless Personal Communications*, vol. 81, no. 1, pp. 253–278, 2015.

[17] M. Q. Rieck and S. Dhar, "A new pruning method for efficient broadcasting in ad hoc networks," *International Journal of Mobile Network Design and Innovation*, vol. 3, no. 4, pp. 209–217, 2011.

[18] G. Călinescu, I. I. Mandoiu, P.-J. Wan, and A. Z. Zelikovsky, "Selecting forwarding neighbors in wireless ad hoc networks," *Mobile Networks and Applications*, vol. 9, no. 2, pp. 101–111, 2004.

[19] C. J. Colbourn and L. K. Stewart, "Permutation graphs: connected domination and steiner trees," *Discrete Mathematics*, vol. 86, no. 1-3, pp. 179–189, 1990.

[20] W. Peng and X.-C. Lu, "On the reduction of broadcast redundancy in mobile ad hoc networks," in *Proceedings of the 1st ACM international symposium on Mobile ad hoc networking & computing*, pp. 129–130, 2000.

[21] R. Rab, S. A. D. Sagar, N. Sakib, A. Haque, M. Islam, and A. Rahman, "Improved self-pruning for broadcasting in ad hoc wireless networks," *Wireless Sensor Network*, vol. 9, no. 02, p. 73, 2017.

[22] W. Peng and X. Lu, "AHBP: An efficient broadcast protocol for mobile ad hoc networks," *Journal of computer science and technology*, vol. 16, no. 2, pp. 114–125, 2001.

[23] M. Sheng, J. Li, and Y. Shi, "Relative degree adaptive flooding broadcast algorithm for ad hoc networks," *IEEE Transactions on broadcasting*, vol. 51, no. 2, pp. 216–222, 2005.

[24] P. Kyasanur, R. R. Choudhury, and I. Gupta, "Smart gossip: An adaptive gossip-based broadcasting service for sensor networks," in *International Conference on Mobile Adhoc and Sensor Systems (MASS), Vancouver, BC, Canada October 9-12*, 2006.

[25] M. A. Spohn and J. Garcia-Luna-Aceves, "Enhancing broadcast operations in ad hoc networks with two-hop connected dominating sets," in *International Conference on Mobile Ad-hoc and Sensor Systems (MASS), Florida, USA October 24-27*, 2004.