2009

Priority Based Routing for Mobile Peer-To-Peer Communications

Swathi Venugopal
San Jose State University

Priority Based Routing for Mobile Peer-To-Peer Communications

CS298 Report
Spring 2009

Submitted by
Swathi Venugopal

Under the guidance of
Dr. Melody Moh
Dept. of Computer Science
San Jose State University

Acknowledgement

I thank my project advisor Dr. Melody Moh for her great support and guidance throughout the research and implementation stages of this project.

I also thank professors Dr. Sami Khuri and Dr. Mark Stamp for taking their precious time to evaluate this project.

Abstract

In a Mobile Peer-to-Peer (MP2P) network, mobile nodes share their resources among one another in a mobile wireless environment. Communication among nodes in MP2P network has become an important area for research due to the significance of its applications. The success of these MP2P applications depends on the number of users in the network, popularity of services offered, quick response and faster access to services. Some services offered could be more popular than others and some peers may contribute more to the network by catering to more requests compared to other peers. In priority based routing mechanism, there is an increase in the priority of a peer with the increase in the number of times it provides services to other peers. The priority of a shared service also increases as the number of requests for that service increases. Also, the mechanism of priority based mobile peer-to-peer routing provides higher priority for traffic destined to high contributing peers and the traffic of popular services, during routing. This would provide high contributing peers quicker response and faster access to services. Hence, this mechanism motivates more users to join the MP2P network and contribute more to the network.

Acronyms

AMPP – Anycast based Mobile Peer-to-Peer

AODV – Ad-hoc On-demand Distance Vector

DSR – Dynamic Source Routing

ID – Identifier

IP – Internet Protocol

MANET – Mobile Ad-hoc Network

M-CAN – Mobile Content Addressable Network

MADPastry – Mobile Ad-hoc Pastry

MP2P – Mobile Peer-to-Peer

NS-3 – Network Simulator – 3

OLSR – Optimized Link State Routing

OSPF – Optimized Shortest Path First

P2P – Peer-to-Peer network

QoS – Quality of Service

TCP – Transmission Control Protocol

Table of Contents

1. Introduction

Mobile Peer-to-Peer (MP2P) network facilitates participating mobile devices to share their resources such as data, bandwidth, storage and computing power in mobile wireless environment. The applications of MP2P communication include instant communication, data distribution and interactive gaming [Persson, 2007]. The major building blocks of MP2P communication are service lookup, peer discovery, routing, privacy and security [Gerla, 2005]. In a MP2P network node seeking a shared object should be able to determine which shared objects are stored in which peer nodes. This service is provided by lookup protocol. Routing protocol facilitates routing information among peer nodes. Hence, efficient lookup and routing protocols form an integral part of MP2P network.

The existing routing protocols used for MP2P communication do not give prominence to factors such as priority of a service, priority of a peer and Quality of Service requirements. Some shared services could be more popular than others and may be frequently utilized by many peers in the network, which determines priority of those services. The traffic of these popular services deserves higher priority over network resources during routing compared to traffic of other services. Also, some peers may contribute more to the network by sharing more number of services or by sharing popular services that are frequently accessed or utilized. These factors determine the priority of those peers. The traffic of these peers deserves higher priority on network resources during routing compared to traffic of other nodes. Hence, priority based routing for MP2P network provides higher priority to traffic of high contributing nodes and popular services in routing compared to the rest of network traffic.

Section 2 describes MP2P network. Section 3 discusses some of the P2P protocols proposed so far. Section 4 describes priority based routing and section 5 evaluates its performance. Finally section 6 concludes priority based routing.

2. Background – MP2P Network

2.1 Overview

A peer-to-peer (P2P) network is a network in which the participant nodes co-ordinate with one another by sharing their resources such as content, service, bandwidth, storage, computing power or a combination of these. Content could be data, media item or a group of items available to the end user as a service. In centralized P2P network a central element manages information about connected peers and the shared resources. Decentralized P2P network is a pure peer-to-peer network where all the nodes have equal status and there is no central element for co-ordination. In hybrid P2P network some nodes are peers where as some nodes are super peers who manage the P2P network.

An overlay can be defined as a subset of nodes in a network that form another network. These overlay nodes may perform different services for the underlying physical network such as data lookups, dynamic routing, and storage or combination of all. Nodes willing to share their resources can form a P2P overlay network. These nodes are also called as peers.

A mobile ad-hoc network (MANET) is a collection of autonomous mobile nodes in a decentralized wireless network. They dynamically self organize themselves to form an arbitrary topology without using any pre-existing infrastructure.

Incorporating peer-to-peer network characteristics in mobile ad-hoc networks can be called as P2P MANET or Mobile P2P (MP2P) network.

The applications of MP2P communication [Persson, 2007] include

- Instant communication involving text, audio and video streaming,
- Data distribution which involves distributing real-time content as well as stored content
- Interactive gaming which includes applications such as real-time interactive games (players react immediately) and turn based strategy games (game is locked after each player's move until the opponent makes his move).

2.2 Characteristics

MP2P network has several characteristics inherited from P2P network and MANET [Kortuem, 2001]. A MP2P network is

- Highly dynamic – The peer nodes can move frequently and independently of one another.
- Self-organizing – Mobile nodes constantly adjust their topology in MP2P network by discovering new communication links.
- Decentralized – Each node in MP2P network is equally important and so nodes are called as peers. No central node exists to control them.
- Infrastructure less – MP2P network does not rely on wired base stations. Hence, it can be deployed in places without existing infrastructure.
- Collaboration – Peers share resources such as storage, content, bandwidth, processing power or combination of all. This collaboration provides high availability and extensibility to MP2P network.
- Fault-tolerant – The malfunction or unavailability of one or more peers may affect the performance of MP2P network. However, MP2P network can still be operational by reconfiguring the network with the help of available peers.

2.3 Challenges

However, incorporating P2P characteristics in MANET poses several challenges too [Persson, 2007 and Mauthe, 2003] as described below:

Wireless Network – The unpredictable characteristics of wireless channel cause wireless network to have less bandwidth, more latency, less connectivity and less stability compared to wired networks. Even if the efficiency of wireless networks is improved to transmit at higher bandwidth, the limited power availability affects the effective throughput.

Constraints of Mobile devices – High mobility of peers causes frequent link failures and packet loss. It also affects data and peer availability. These devices have limited battery power and are smaller in size compared to stationary devices. Hence, they tend to have less processing power, less memory, limited power supply, smaller display screens, missing or inefficient input devices and less intensive security measures.

Automatic configuration – Due to ad-hoc nature of MP2P network, nodes must dynamically and automatically decide whether to participate in a P2P overlay. The constraints of mobile devices affect them during their participation in P2P overlay.

Addressing – Peers cannot be guaranteed with the availability of IP address, as they cannot always access DNS servers in mobile ad-hoc networks. Also not all mobile devices support IP networking and thus may not have IP addresses.

Peer and Resource Discovery – Unpredictable physical mobility of peers makes discovering peer as well as shared resource a challenge. Although ad-hoc network handles peer discovery, efficient resource discovery in MANET is the responsibility of MP2P system.

Decentralized coordination – MANET cannot have a single dedicated node for coordinating peer behavior.

Consistency – In order to provide high availability peers tend to maintain local copy of shared data object. This replication allows copies of a shared object to be updated independently, which in mobile ad-hoc environment, may lead to inconsistency.

Timeliness - Data might be shared across a group of peers that never meet all at the same time. If any service in an ad hoc system depends on the interaction between two peers who may not meet frequently, then this situation can lead to slow propagation of service in the network.

Scalability – In a heterogeneous MP2P network, a larger number of control messages to co-ordinate different kinds of peers might limit the scalability of MP2P network.

Privacy – Every peer has the right to control the services it shares and control the use of its personal information. MP2P network must protect a peers' anonymity whenever desired. A MP2P network must not only prevent spying and monitoring, but allow peers to control what information can be disclosed, to whom, and when.

Transparency – Concurrent access of services by multiple peers and mobility of peers should be kept transparent.

Security – The physical location, content and communications of peer should be protected in MP2P network. Adopting traditional security schemes like firewalls may not be feasible due to the limited storage, battery and processing power of mobile nodes. Also these traditional security measures might hinder the peer-to-peer communications.

2.4 Building Blocks

The major building blocks of MP2P communication are service lookup, peer discovery, routing, privacy and security [Gerla, 2005]. The factors to be considered for efficient service lookup and routing protocols are dependent on the peer's application requirements and network parameters. The requirements of the peer's application that must be considered are:

- Query rate – The amount and distribution of queries in the network
- Replication Rate – The probability of finding a given data object in a network
- Popularity – The query statistics for a particular object i.e. the demand for a particular data object.
- Scale of objects – The size and statistics of the object population. In particular it defines the average number of objects per node and its variance.
- Quality of service requirements – Delivery ratio and latency, routing consistency etc.

Some of the network related factors to be considered are:

- Mobility scenario – The speed, obstacles, and propagation models of the network.
- Scale of nodes – The size of the join/leave/failure statistics of the participating nodes.
- Extent of network partitioning and merging
- Network density – Average number of nodes per space unit
- Size and speed of nodes
- Type of the network – Pure/Heterogeneous networks, Ad-hoc/Hybrid (network with both ad-hoc and infrastructure nodes) network
- Privacy and security of node as well as data traffic

3. Related Study

In order to facilitate routing in MP2P network, two design options have been considered – layered approach and integrated approach. In layered approach P2P protocol is implemented in a peer overlay network that uses existing MANET routing protocols in the underlying physical network. In integrated approach P2P protocol is integrated with existing multi-hop MANET routing protocol.

Thus, layered approach induces multiple layer redundancy and duplication in terms of messages and communication between nodes. But, layered approach avoids cross layer dependency. This provides scalability to P2P protocols and routing protocols. Integrated approach reduces routing overhead compared to layered approach. But, it may affect the scalability of P2P protocols and routing protocols due to cross layer dependency. Which approach among the two is better?, is still an open issue.

3.1 Layered Approach

In layered approach, many protocols such as M-CAN [Peng et al, 2004], Chordella [Stoicay et al., 2003 and Hofstätter at el., 2008], Pastry [Rowstron, 2001], etc., have been proposed for efficient communication among peers in MP2P network.

3.1.1 M-CAN

Protocol Description:

According to this protocol, resources should be well organized when they join the system. In this protocol, every peer is assigned a unique ID randomly. IDs are also assigned to shared resources according to their title and content.

Some nodes are selected as super nodes, because of their stronger capacity and more reliable connections. They also manage a range of content IDs separately. Content Addressable Network manages these super nodes. Every node will be registered on one or more super nodes according to the IDs of its shared contents. Every super node is registered on itself. Figure 3-1 depicts the structure of M-CAN [Peng et al, 2004].

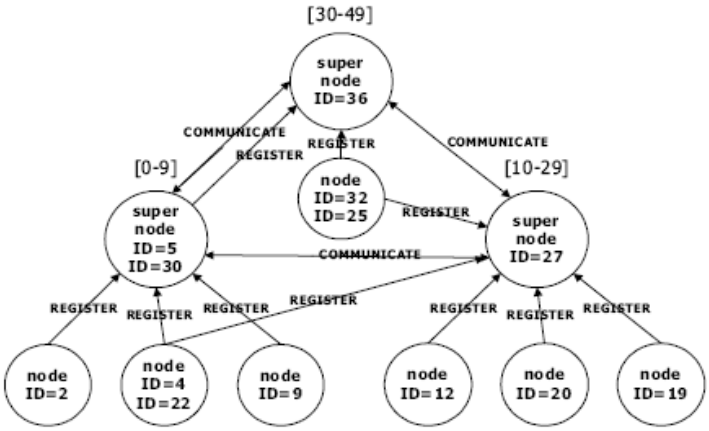

Figure 3-1: Example for Structure of M-CAN. [Peng et al, 2004]

The join process begins when a new node broadcasts first JOIN request, containing one of the shared content IDs to all the neighboring nodes. The new node will only be registered on the super node that replies to this JOIN request first, because, most probably the new node has a more reliable and stable connection with the first replied super node. This join process will be repeated until all the shared resources have been covered.

In order to avoid overloading the super node a limit is set on the maximum number of nodes that any super node can manage, say ‘n’. If a super node manages more than ‘n’ nodes then it checks its directory and divides the nodes into two equal sized sub groups.

When an ordinary node leaves M-CAN [Peng et al, 2004], only the directory of its super node(s) is modified. When a super node leaves the network, one of its neighbor super nodes extends its ID space to cover the member nodes of missing super node. All the member nodes registered on the missing super node, register themselves on this new super node. A set of super nodes that cover the whole ID space and the ordinary nodes registered on them form a group. Communication between groups is supported by communicate nodes, which are nodes at the edge of a group and have a good connection with nodes in other groups.

Source node finds the ID of the desired resource first. Then it sends a request to source super node for the desired resource ID. The source super node routes this request to the destination super node

in its group. Then the destination super node launches a lookup process locally for the desired resource ID. If the destination node (node containing desired data) is registered on the destination super node, then destination super node returns the address of the destination node to the Source node. Otherwise, the destination super node will broadcast the request to other groups through communicate nodes and the lookup process will be triggered in other groups. Figure 3-2 demonstrates the lookup process in M-CAN [Peng et al, 2004].

After the Source node gets the address of destination node, it would try to communicate with the destination node directly. Only when the direct access is impossible, the source node will communicate with destination node with the help of other nodes. If source node does not receive a response within a predefined period of time, it is assumed that the desired data does not exist in the whole peer community. The communication among the peer nodes is achieved through the routing protocols available at the network layer.

When network has higher number of peers and higher number of lookup requests, M-CAN [Peng et al, 2004] has lesser average request latency compared to centralized directory lookup and flooding.

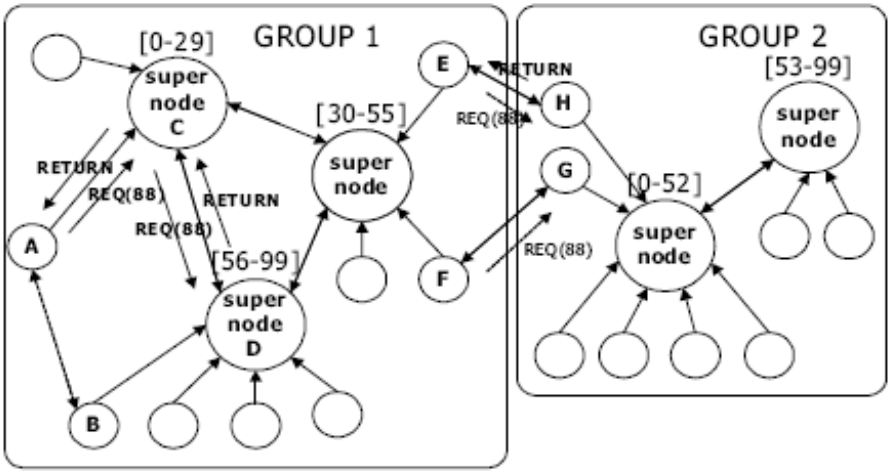

Figure 3-2: Example of Lookup process in M-CAN. [Peng et al, 2004]

Analysis:

M-CAN [Peng et al, 2004] maintains shared resources in well-organized manner, which reduces lookup request and response latency under well-distributed content and normal mobility environment.

However, if the shared contents are not uniformly distributed among different M-CAN groups, then the lookup request and response latency might increase for nodes of the group who do not have sufficient shared contents in them. If nodes have high mobility then messages associated with the frequent reorganization of shared resources increases network traffic. Hence, in case of network with lower number of nodes and higher node mobility, maintaining M-CAN might be an unnecessary overhead. Finally, the demand for each shared resource may not be same. Popular or most sought after resources may have more lookup requests and responses. M-CAN does not give any priority to these lookup requests and responses to reduce their latency.

3.1.2 Chordella

Protocol Description:

Chord [Stoicay et al., 2003] is a P2P protocol for efficiently locating a node that stores a desired data item. Each node is assigned a unique identifier and each shared object is also assigned a unique key. The peers are ordered on a circle of identifiers from 0 to 2m – 1 to form a Chord Ring where ‘m’ is the number of bits used for node/key identifier. Key ‘k’ is assigned to the first node whose identifier is equal to or follows (the identifier of) k in the Chord ring. Each node ‘n’ maintains a routing table called ‘Finger Table’. Each entry in the table for a node contains its identifier, IP address and port number. The entry ‘i’ in the table at node ‘n’ contains the identity of the first node ‘s’ that succeeds ‘n’ by at least 2i-1 on the identifier circle, i.e., s = successor (n+2i-1), where 1 <= i <= m (and all arithmetic is modulo 2m) [Stoicay et al., 2003]. The first finger of ‘n’ is the immediate successor of ‘n’ on the identifier circle; conveniently referred as the successor. The previous node on the circle is referred to as predecessor.

A peer node that desires a shared object obtains its key from distributed hash table, where the mapping of shared object and its key is stored. In a simple lookup process, source peer node looks for key in its every successor. If the value of key is equal to or less than the successor’s identifier,

then that successor is chosen as the destination peer. In a scalable lookup process as shown in Figure 3-3, the source peer node looks for the key in its immediate successor. If it is not present, then source peer node determines its closest preceding node by checking every entry of its finger table. The closest preceding node is the node whose identifier is equal to or closest to the key. Then the scalable lookup is repeated in this closest preceding node and so on.

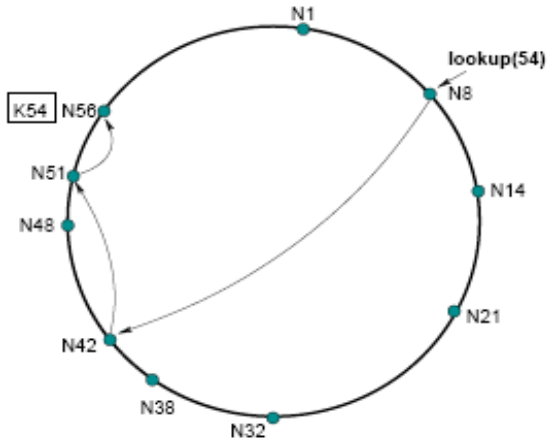

Figure 3-3: Scalable lookup in Chord ring. [Stoicay et al., 2003]

Nodes stabilize their stored information by periodically updating their finger table, successor and predecessor information. When a new node joins the Chord ring, it is aware of its predecessor. If the identifier of new node's predecessor's immediate successor say 'S' is greater than the identifier of new node, then new node sets that 'S' as its successor. Then the new node copies all keys less than or equal to its identifier from its successor 'S'.

A node leaves the Chord [Stoicay et al., 2003] ring on two scenarios – node failure and voluntary node departure. Each node maintains a successor list containing its first 'r' successors. During node failure, if node n notices that its successor has failed, it replaces it with the first live entry in its successor list and reconciles its successor list with its new successor. During voluntary node departure, node 'n' that leaves may notify its predecessor 'p' and successor 's' before leaving. Node 'n' sends its predecessor information to 's', and the last node in its successor list to 'p'. Node 's' and 'p' update their successor and predecessor information accordingly. Also node 'n' may transfer its keys to its successor before it departs.

Chordella [Hofstätter at el., 2008] classifies mobile nodes into two groups – leaf nodes and super peers. Leaf nodes are energy constraint mobile nodes with low computing power. Super peers are nodes with substantial resources and more reliable network connection [Hofstätter at el., 2008]. Super peers form Chord [Stoicay et al., 2003] ring and they maintain point-to-point connection with leaf nodes. The number of super peers is dynamically adjusted to minimize network cost. Mobile nodes are promoted to super peer or demoted to leaf node state depending on their capacity and network requirements. Reference to popular content is cached in nodes to minimize lookup latency. Figure 3-4 shows Chordella P2P network topology.

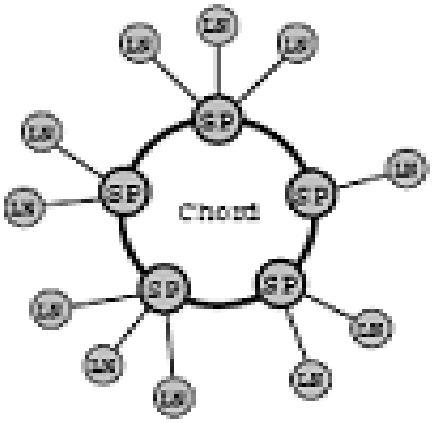

Figure 3-4: Chordella P2P network. [Hofstätter at el., 2008]

In order to ensure equal load on all the super peers, Chordella follows Piggyback load balancing algorithm [Zoels, 2007]. In this technique, every message exchanged between two super peers is piggybacked with the current load of the sending super peer. Underlying network layer routing protocols support the communication among overlay nodes.

Analysis:

This protocol [Hofstätter at el., 2008] optimizes lookup latency by caching reference to popular content. It aims to provide equal load on all peer nodes by altering their super peer – leaf node status; by adjusting the number of leaf nodes managed by a super peer.

However, this protocol does not aim to optimize the routing latency. It does not provide priority to peers that contribute more to network than others. In case of high network density or high query

rate separate communication in the overlay network and underlying network simply increases the routing overhead.

3.1.3 Pastry

Protocol Description:

Pastry [Rowstron, 2001] is a generic peer-to-peer object location and routing scheme, based on an overlay network of nodes connected to the Internet. Pastry is completely decentralized, fault-resilient, scalable, and reliable.

Each node in the Pastry peer-to-peer overlay network is assigned a 128-bit node identifier (node ID) [Rowstron, 2001]. This ID identifies a node's position in a circular node ID space, which ranges from 0 to 2128 - 1. Each Pastry node maintains a routing table, a neighborhood set and a leaf set. The routing table at every node contains $\lceil log_{2^b} N \rceil$ [Rowstron, 2001] rows with 2b-1 entries in each row. The 2b-1 entries at row 'n' of the routing table, each refer to a node whose node ID shares the present node's node Id in the first n digits, but whose (n + 1)th digit has one of the 2b-1 possible values other than the (n + 1)th digit in the present node's id [Rowstron, 2001]. The neighborhood set M at every node contains the node IDs and IP addresses of the |M| nodes that are closest (according the proximity metric) to the local node [Rowstron, 2001]. The leaf set L contains nodes with the |L|/2 numerically closest larger node IDs, and the |L|/2 nodes with numerically closest smaller node IDs, relative to the local node's ID.

Given a message and key, a Pastry [Rowstron, 2001] node efficiently routes the message to the node with a node ID that is numerically closest to the key, among all currently live Pastry nodes. The expected number of routing steps is O(log N), where N is the number of Pastry nodes in the network. The node first checks to see if the key falls within the range of node IDs covered by its leaf set. If so, the message is forwarded directly to the destination node, namely the node in the leaf set whose node ID is closest to the key. If the leaf set does not cover the key, then the routing table is used and the message is forwarded to a node that shares a common prefix with the key by at least one more digit. If the appropriate entry in the routing table is empty or the associated node is not reachable, then the message is forwarded to a node that shares a prefix with the key at least as long as the local node, and is numerically closer to the key than the present node's ID.

When a node joins or leaves the circular identifier space, the affected routing tables, neighborhood sets and leaf sets are updated appropriately. Pastry minimizes the distance messages travel, using scalar proximity metric like the number of IP routing hops or geographic distance.

If the network faces arbitrary node failures or malicious node behaviors then Pastry can afford to adopt random routes based on average route delay statistics. Pastry prevents node isolations by updating neighborhood information through periodic IP multicast.

Analysis:

Pastry [Rowstron, 2001] is an efficient P2P protocol which routes at O(log N) steps in a self-organizing overlay network. Unlike Chord [Stoicay et al., 2003], which maintains only a successor list, Pastry maintains neighborhood set, which contains information on all the neighbors based on proximity. Pastry follows layered approach where routing between two peer nodes may involve communication among several nodes in the underlying network. It is an efficient lookup protocol but does not provide priority to highly contributing peers or popular services. Since any routing protocol can be used in the underlying network, Pastry does not contribute to efficient P2P routing.

3.2 Integrated Approach

Some of the MP2P protocols that follow integrated approach are Mobile Ad-hoc Pastry (MADPastry) [Rowstron et al., 2001 and Zahn et al., 2005], Anycast based MP2P routing (AMPP) [Cheng et al., 2005] etc.

3.2.1 AMPP

Protocol Description:

AMPP [Cheng et al., 2005] routing protocol aims to reduce routing overhead and optimize lookup service by integrating anycast AODV routing protocol with Chord protocol [Stoicay et al., 2003] in network layer of MANET.

Every peer node in the network has unique identifier. Peers offering same service form an anycast group with each group having unique anycast group ID. Thus the P2P network may contain both

anycast peer nodes and non-peer unicast nodes. All members of each anycast group share the same anycast group ID. The protocol assumes that the number of anycast groups is steady in a MANET. Whenever a mobile node joins or leaves an anycast group, its anycast group ID is updated correspondingly. Each service is identified by unique service key. Chord [Stoicay et al., 2003] technology maps the service key onto an anycast group providing that particular service. Anycast routing chooses the best receiver from the anycast group.

AMPP [Cheng et al., 2005] implements anycast routing by extending AODV routing protocol to hold service key and anycast group ID information. If the source node is unicast node, it simply sends desired service key to default anycast group. The member(s) of default anycast group locate the service using Chord [Stoicay et al., 2003] protocol. If the source node is anycast node, it locates the service using Chord lookup protocol. The anycast nodes use anycast routing at the IP layer to locate the closest service provider.

In the IP layer, if an intermediate node is a unicast node it simply re-broadcasts P2P traffic to its neighbors. If the intermediate node is an anycast node, if it has the service key, it responds with destination IP address and corresponding anycast group ID to source node. If the intermediate anycast node does not have the service key, it checks its group's finger table. It selects an anycast group say S that is closest to the key using Chord's lookup algorithm [Stoicay et al., 2003]. Then it checks its routing table entries for nodes belonging to anycast group S, to check whether there exists a route for this communication. If there are many routes, it will choose the route with minimum hop count and the corresponding destination as the next hop node. If there does not exist a route to any node in anycast group S, it will re-broadcast the service request to anycast group S.

The source node selects the closest route to destination anycast group ID from its routing table. Figure 3-5 depicts AMPP routing.

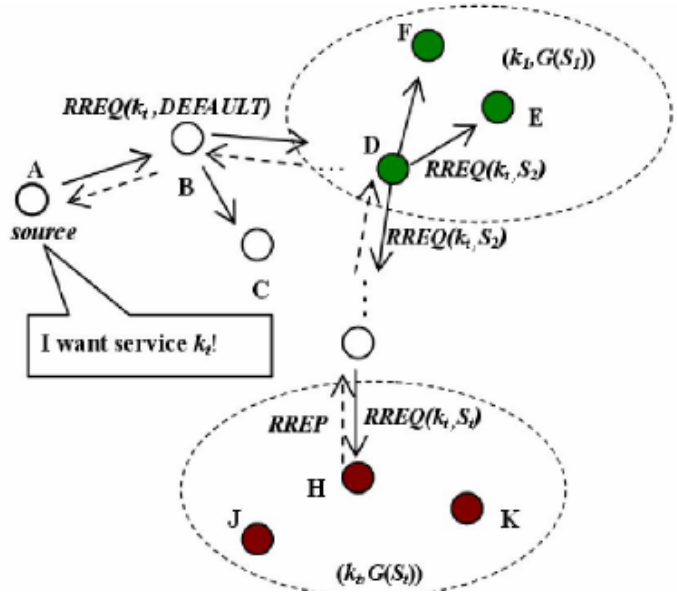

Figure 3-5: AMPP routing. [Cheng et al., 2005]

Analysis:

AMPP [Cheng et al., 2005] organizes services into anycast groups. It assumes that the number of such groups in MANET is steady. However, if new services are frequently introduced and older services are removed from the network, then managing the reorganization of services as anycast groups is a challenge. If peers frequently vary the services that they share or if they are highly mobile then updating their corresponding anycast group ID is also a challenge. These limitations may affect the network performance due to increased number of control messages. Since AMPP uses the AODV routing protocol at the IP layer, it assumes the availability of IP address for every mobile node in the anycast group. If the network contains heterogeneous mobile devices, not all devices support IP networking. Then implementing AMPP in such a scenario is a challenge.

3.2.2 MADPastry

Protocol Description:

MADPastry [Zahn et al., 2005] integrates Pastry [Rowstron et al., 2001] with AODV routing protocol. It forms physical clusters of peer nodes whose identifiers are close to a predetermined key called ‘landmark key’. So nodes that are close to each other in overlay network due to their identifiers are also likely to be close to each other in the underlying physical network.

A landmark key is simply an overlay ID. Landmark keys should be chosen so that they divide the overlay ID space into equal-sized segments [Zahn et al., 2005]. Landmark node is the node whose identifier is currently closest to landmark key. When any landmark nodes fails or resigns, another node whose overlay ID is now closest to the landmark key is assigned the status of landmark node. Landmark node periodically transmits beacons. Only the nodes in its cluster forward these beacons to avoid routing overhead due to broadcasting. Nodes that listen to these beacons form a landmark list consisting of landmark node's ID and the distance to it as given by the hop count of the beacon. Nodes periodically examine their landmark list to determine whether they have moved closer to a new landmark node, i.e. into a new cluster. If so, a node will assign itself a new random overlay ID with its new cluster's overlay ID prefix, resign from the overlay network with its old ID, and rejoin the overlay network with its new ID.

MADPastry [Zahn et al., 2005] maintains three routing tables – AODV routing table, MADPastry routing table and leaf set. The AODV routing table maintains physical routes to destination nodes. MADPastry routing table consists of $\lceil \log_{2^b} K \rceil$ [Zahn et al., 2005] rows where K is the number of landmark keys.

In MADPastry, when a node receives a request packet, there are two possibilities. If the node is destination node of an overlay hop, then it needs to determine the next overlay hop. It performs standard Pastry routing by consulting its MADPastry routing table to find a node that would increase the matching key prefix by one or its leaf set to find a node that is numerically closer to the key than the current node is [Zahn et al., 2005].

If the node is an intermediate node on the physical path of an overlay hop then the node consults its AODV routing table to determine the next physical hop towards the destination. To minimize the routing traffic, intermediate node inspects the destination of the overlay hop [Zahn et al., 2005]. If the intermediate node's own overlay ID is already numerically closer to the packet's key than that of the overlay hop's actual destination, then the intermediate node considers the current overlay hop as completed [Zahn et al., 2005]. It then selects from its MADPastry routing table or leaf set the next overlay hop. When physical route to an overlay hop is unavailable, if the intermediate node is already in destination cluster, it simply broadcasts overlay packet within its

cluster. If the intermediate node is not in the destination cluster, it uses AODV protocol to discover the route to destination node.

Analysis:

MADPastry [Zahn et al., 2005] reduces routing overhead by integrating overlay routing and physical routing. However, it does not consider the possibilities of a node providing multiple services. It does not provide any priority to popular services and peers that contribute more to P2P network.

4. Priority Based MP2P Routing

4.1 Overview

Priority based MP2P routing follows layered approach as it avoids cross layer dependency and allows lookup and routing protocols to be scalable. The priority of a peer node is referred as Node Priority and priority of offered service is called Service Priority. The lookup table, which usually maintains the service identifier – service provider node mapping, is expanded with information such as Service Priority, and priority of service provider node. The Node Priority of a peer is determined by the number of times the peer has catered to service requests from the other peers. The number of requests raised in the network for a service determines its Service Priority.

For all the packets originated at a node, they are routed in the descending order of Node Priority of the peers to whom the packets are destined. For all the packets received at a node, if the packets are request messages such as lookup request, request for service etc, then they are processed in the descending order of Node Priority of requestor nodes (source nodes of the packets). If the packets received are non-request messages, then they are processed in the descending order of Node Priority of destination nodes (nodes to whom the packets are destined). During prioritization of packets, if the Node Priority of any two packets is same, then they are prioritized based on the priority of service to which they belong.

The priority of a service increases with every request issued for that service. The priority of a peer node also increases with node catering to every request from its peers. The validity of Service Priority and Node Priority information in lookup table is maintained by the periodic update of lookup table by lookup protocol.

Since nodes provide higher priority to traffic of higher priority nodes and popular services, the time taken to receive a response must be lower for higher priority peers.

4.2 Detailed Design

4.2.1 Determine priority information during lookup

Every peer node maintains a lookup table with information such as identifier of a service, priority of that service which is the number of requests received for that service, address of the service provider node, and priority of service provider node which is the number of times that node has responded for a service request.

Peer node looks for service provider:

When a peer node wishes to get a service offered by any other peer node, it is expected to know the service identifier of the desired service. Then peer node checks its lookup table to determine the service provider node. If the information is available in its lookup table, it increases the priority of desired service and then sends a service request to service provider node. If the information is not available in lookup table, then peer node broadcasts a lookup request for desired service.

```
When peer wishes to get any service offered by other peer nodes
{
    Check its lookup table to find the address of node that offers desired service
    If the information is available in lookup table
    {
        Update Lookup Table – Increment the Service Priority or number of service requests for
        the desired service
        Send Service Request to Service Provider node with Request Identifier
     }
    Else          //information is not available in Lookup Table
   {
        Broadcast Lookup Request with unique Request Identifier.
    }
}
```

Peer node receives a Lookup Request for a service provider:

All the nodes that receive lookup request update service priority for requested service in their lookup table. They send lookup response to lookup requester node if they are offering the desired service or if they know the node that provides desired service through their lookup table. This lookup response contains information of node that provides desired service and priority of desired service.

```
When a peer node receives Lookup Request
{
    If the Lookup Request is new
    {   Process Lookup Request to determine the desired Service
        If the node is offering the service  // the node that received Lookup Request offers that service
        {
                Update Lookup Table – Increment the Service Priority or number of service requests for the desired service
                Send Lookup Response to source node
        }
        If the node is NOT offering the service
        {       If the desired service has an entry in node's Lookup Table
                {
                        Update Lookup Table – Increment the Service Priority or number of service requests for the desired service
                        Send Lookup Response to source node with address of Service Provider node and priority of service.
                }
                Else
                   Forward the request
        }
    }
}
```

Peer node receives service provider information in Lookup response:

When the source node receives the first lookup response, it updates service priority for requested service in its lookup table. Then it sends a service request to service provider, requesting the beginning of communication.

When source node receives Lookup Response
{
    If the Lookup Response is received for the first time
    {
        Update Lookup Table – Increment the Service Priority or number of service requests for the desired service
        Send Service Request
    }
}

Peer node requests service from service provider:

When service provider receives service request, it sends service response to source node indicating the initiation of communication. It updates service priority and node priority in its lookup table.

When service provider node receives Service Request
{
    Process request to determine the requested service
    If a Lookup Request has not been received earlier corresponding to this Service Request
    {
        Update Lookup Table – Increment Service Priority i.e. no of service requests
    }
    Update Lookup Table – Increment Service Provider Priority i.e. number of times the node responded for a service request
    Send Service Response to source node
}

Peer node receives service from service provider:

When source node and the intermediate nodes receive service response from the service provider they update the corresponding node priority in their lookup tables.

When service requester receives Service Response, time taken to set the communication is calculated which would vary depending on the priority of service and priority of service requestor

When service requester receives Service Response

{

// Communication between sender and receiver is established

Update Lookup Table – Increment Service Provider Priority i.e. number of times the node responded for a service request

Calculate the time taken to receive the response

}

4.2.2 Use priority information during routing

Usually every packet received or every packet to be transmitted is processed in the order of its arrival at the network layer of a node. However, priority based routing attempts to provide higher priority for packets belonging to high contributing peers and popular services over other packets. Hence, all the packets originated at a node are routed based on the Node Priority of destination nodes of the packets. All the request packets such as lookup request or service request received at a node are processed based on the Node Priority of source nodes. If the packets received are non-request messages, then they are processed based on the Node Priority of destination nodes of packets. If the Node Priority of any two packets is same, then they are prioritized based on the priority of service to which they belong.

Peer node receives a packet:

On receiving a packet:

{

Store the packet in incoming queue.

Repeat until incoming queue is empty

{

Process every packet in the queue to obtain the source address of packet from its IP header.

If the source address does not have an entry in Lookup Table, that packet is not given any priority.

Determine the packet of highest priority

{

Use Lookup table to determine the priority of source node and the service to which the packet belongs.

If there is more than one packet belonging to the highest priority node or if there are packets belonging to nodes of same priority then determine the packet that belongs to service of highest priority. This packet is chosen as the packet of highest priority.

If packets belong to nodes of same priority and services of same priority, then the first entered packet in queue is chosen as the packet of highest priority.

}

If I am the destination node

{

Accept the packet of highest priority

}

Else

{

Forward the packet of highest priority

}

}

}

Peer node transmits a packet:

While transmitting a packet

{

Store the packet in outgoing queue.

Repeat until outgoing queue is empty

{

Process every packet in the queue to obtain the destination address of packet from its IP header.

If the destination address does not have an entry in Lookup Table, that packet is not given any priority.

Determine the packet of highest priority

{

Use Lookup table to determine the priority of destination node and the service to which the packet belongs.

If there is more than one packet destined to the highest priority node or if there are packets destined to nodes of same priority then determine the packet that belongs to service of highest priority. This packet is chosen as the packet of highest priority.

If packets are destined to nodes of same priority and belong to services of same priority, then the first entered packet in queue is chosen as the packet of highest priority.

}

Transmit the packet of highest priority to its destination node

}

}

5. Performance Evaluation

Based on the analysis in the table 8-1 in Appendix, the network simulator NS-3 (version 3.3) has been chosen to simulate priority based routing. Priority based routing mechanism uses a simple lookup mechanism to collect priority information. It is implemented over Optimized Link State Routing [Jacquet et al., 2001] protocol, which is used for routing packets.

5.1 Simulation Environment

20 mobile nodes follow Random direction 2-dimentional mobility model. Each node offers one or more services as shown in Table 5-1.

Nodes	Services Offered
0	Movie-GodFather-II
1	Movie-Titanic
2	Movie-Terminator-I
3	Movie-Speed
4	Book-Earth is Flat
5	Book-Twilight
6	Book-HarryPotter
7	Book-KiteRunner
8	Game-Poker, Video-ObamaSpeech
9	Game-WebCarRace, Video-HannahMontana
10	Game-Scrabble, Game-Solitaire
11 – 14	Video-Friends
15 – 19	MobiSkype

Table 5-1: Nodes and the services they offer

The response time for a node is calculated as the time difference between the moment the node sent lookup request or service request and the moment the node received service response and data packets. So, the response time calculated for a node includes the propagation delay as well as the queue delay involved in transmission.

5.2 Simulation Results

5.2.1 Case 1: Nodes have different priority

Traffic is designed such that nodes 0 to 5 dynamically gain higher priority during the initial 10 seconds of simulation, compared to nodes 12 to 17. Lower priority nodes and higher priority nodes transmit P2P traffic at the rate of 1.6Mbps during the next 20 seconds. The traffic pattern is as shown in Table 5-2.

Services Requested and Delivered for P2P traffic	Destination Nodes	Node Priority
Game-Poker, Video-ObamaSpeech	0, 12	2, 0
Game-WebCarRace, Video -HannahMontana	1, 13	3, 0
Game-Scrabble	2, 14	3, 0
Game-Solitaire	3, 15	1, 0
Game-Poker	4, 16	4, 0
Game-WebCarRace	5, 17	1, 0

Table 5-2: P2P traffic where nodes have different priority

The average response time of high priority nodes and low priority nodes during normal routing and during priority based routing are compared. The average response time of high priority nodes is expected to be lower compared to that of low priority nodes during priority based routing.

Scenario 1: Low speed mobility at 0.5 m/s

All nodes move at constant low speed of 0.5 m/s in random directions. Example: People moving with mobile devices in crowded downtown areas.

Priority of nodes does not influence their average response time during normal routing (Figure 5-1a). However, during priority based routing, high priority nodes have lower response time

compared to low priority nodes (Figure 5-1b). The average response time of high priority nodes has decreased in the range 4% to 35% from normal routing to priority based routing (Figure 5-1c).

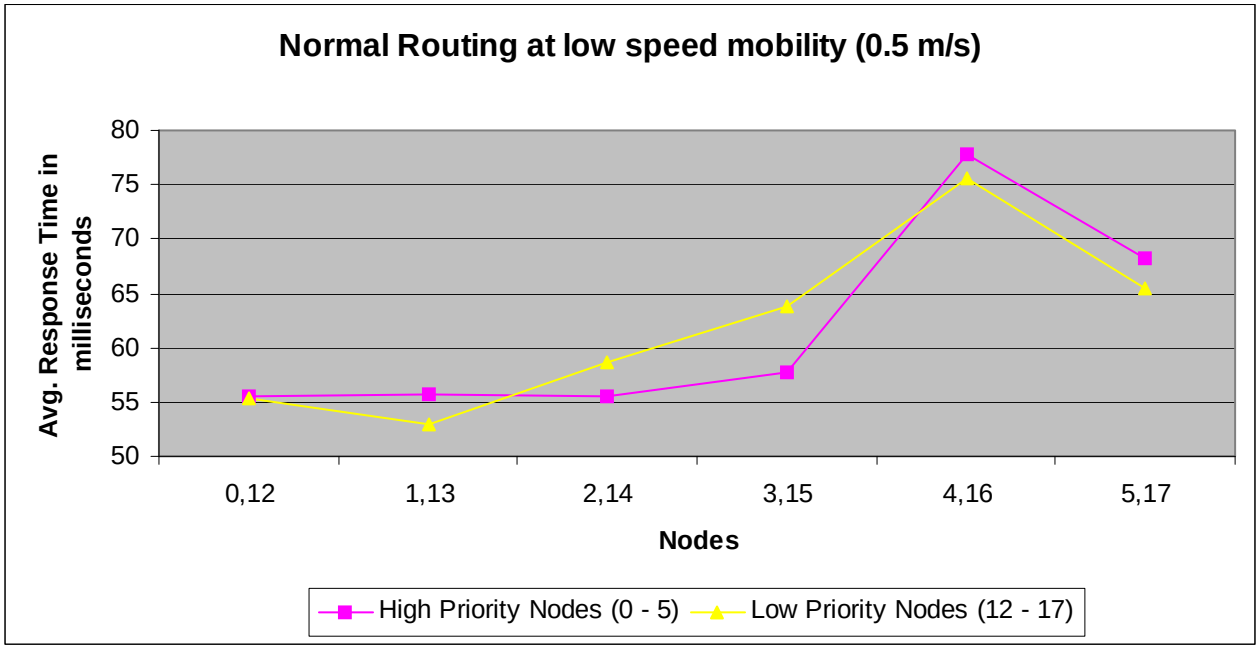

Figure 5-1a: Average response time during normal routing at low speed mobility

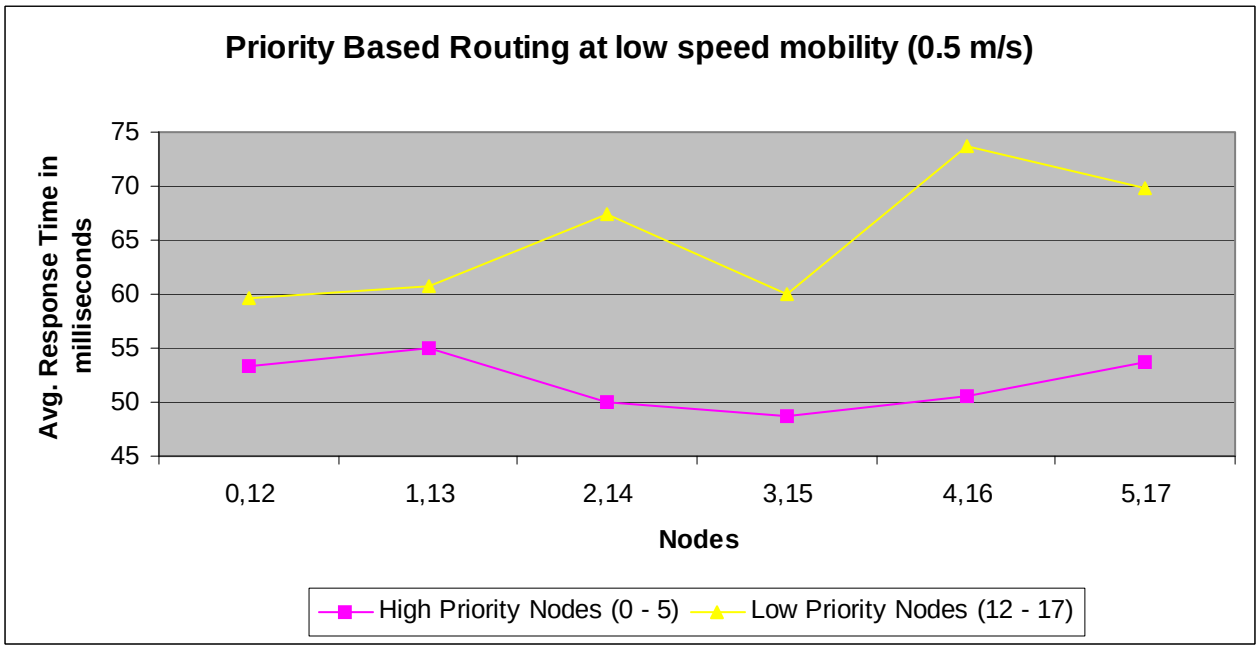

Figure 5-1b: Average response time during priority based routing at low speed mobility

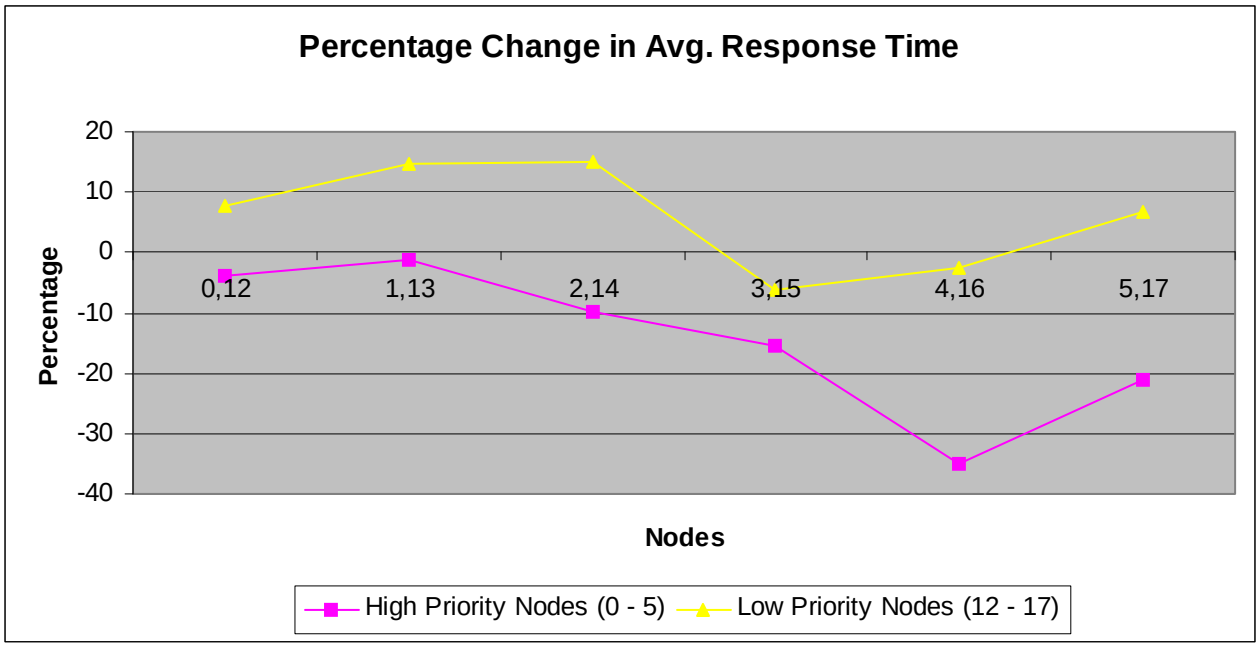

Figure 5-1c: Percentage change in average response time from normal routing to priority based routing at low speed mobility

Scenario 2: High Speed Mobility at 100 m/s

All nodes move at constant high speed of 100 m/s in random directions. Example: People with mobile devices traveling in high-speed vehicles such as buses or trains.

Priority of nodes does not influence their average response time during normal routing (Figure 5-2a). Majority of high priority nodes have lower response time compared to low priority nodes during priority based routing (Figure 5-2b). The average response time of majority of high priority nodes has decreased in the range 1% to 10% from normal routing to priority based routing (Figure 5-2c).

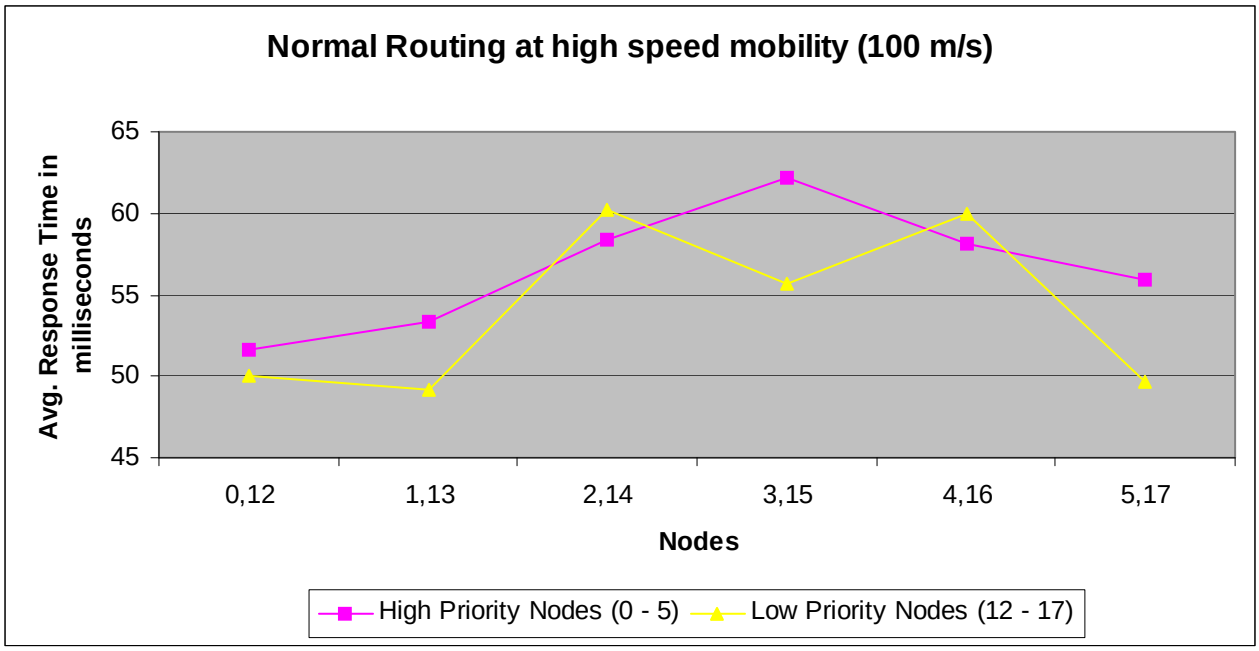

Figure 5-2a: Average response time during normal routing at high-speed mobility

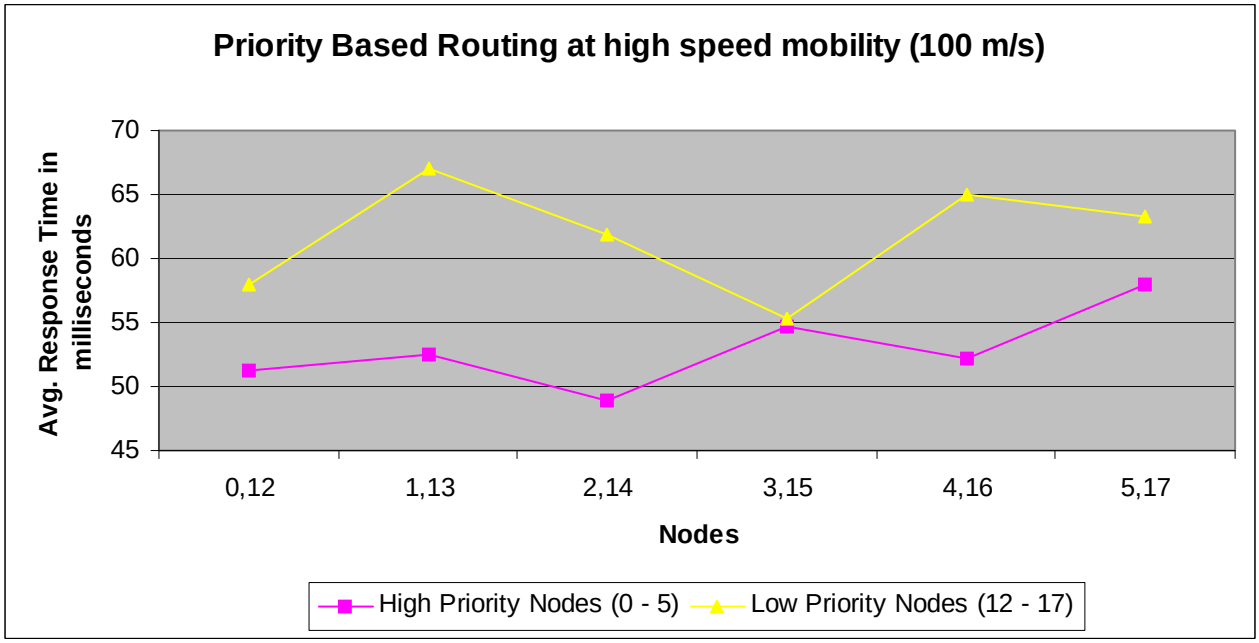

Figure 5-2b: Average response time during priority based routing at high-speed mobility

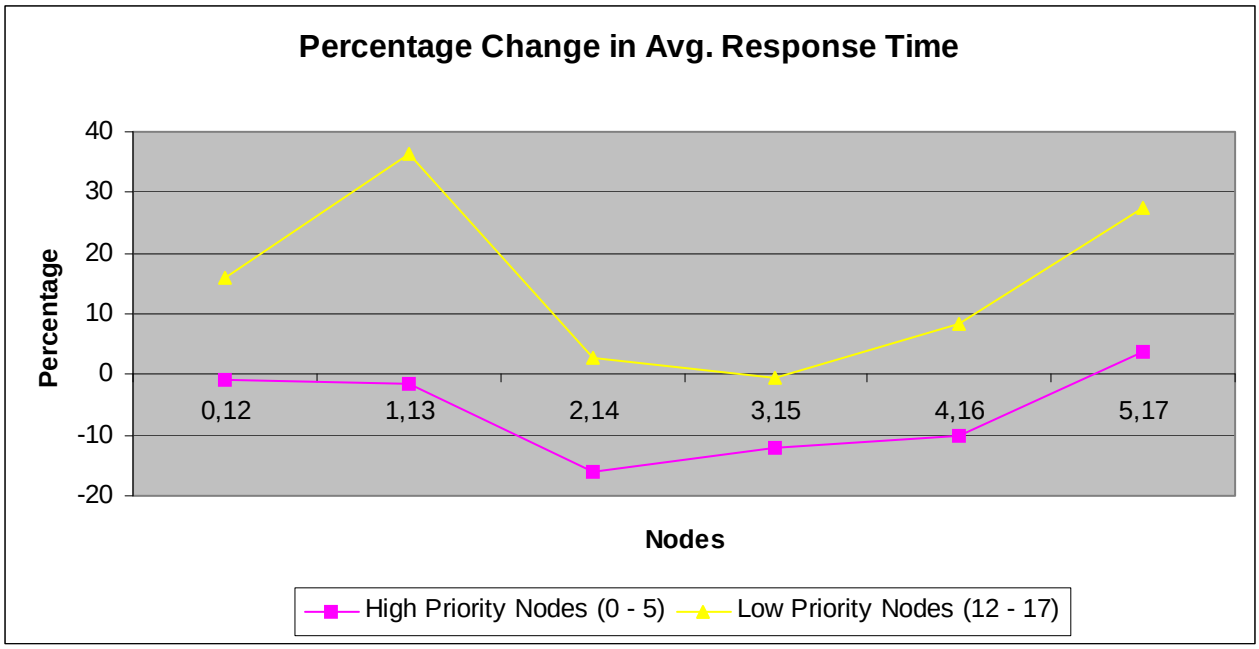

Figure 5-2c: Percentage change in average response time from normal routing to priority based routing at high-speed mobility.

Scenario 3: Random Speed Mobility with speed between 1 m/s and 50 m/s

All nodes move at random speed (1m/s – 50 m/s) in random directions.

Priority of nodes does not influence their average response time during normal routing (Figure 5-3a). Majority of high priority nodes have lower response time compared to low priority nodes during priority based routing (Figure 5-3b). The average response time of majority of high priority nodes has decreased in the range 5% to 30% from normal routing to priority based routing (Figure 5-3c).

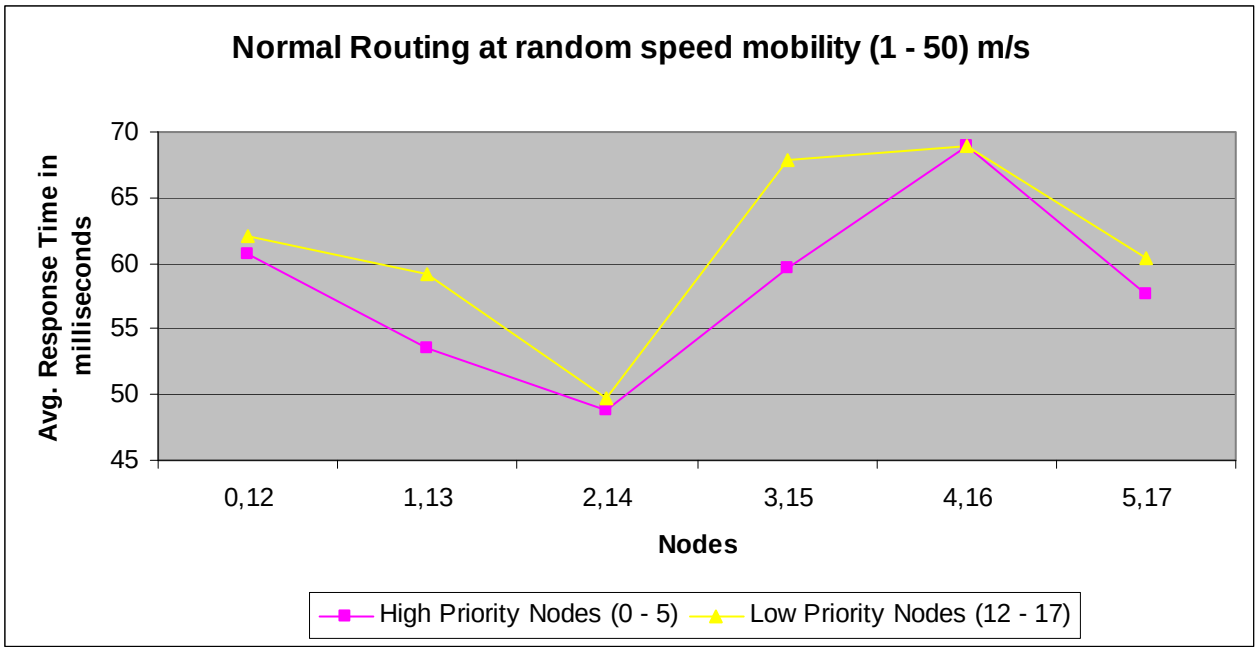

Figure 5-3a: Average response time during normal routing at random speed mobility

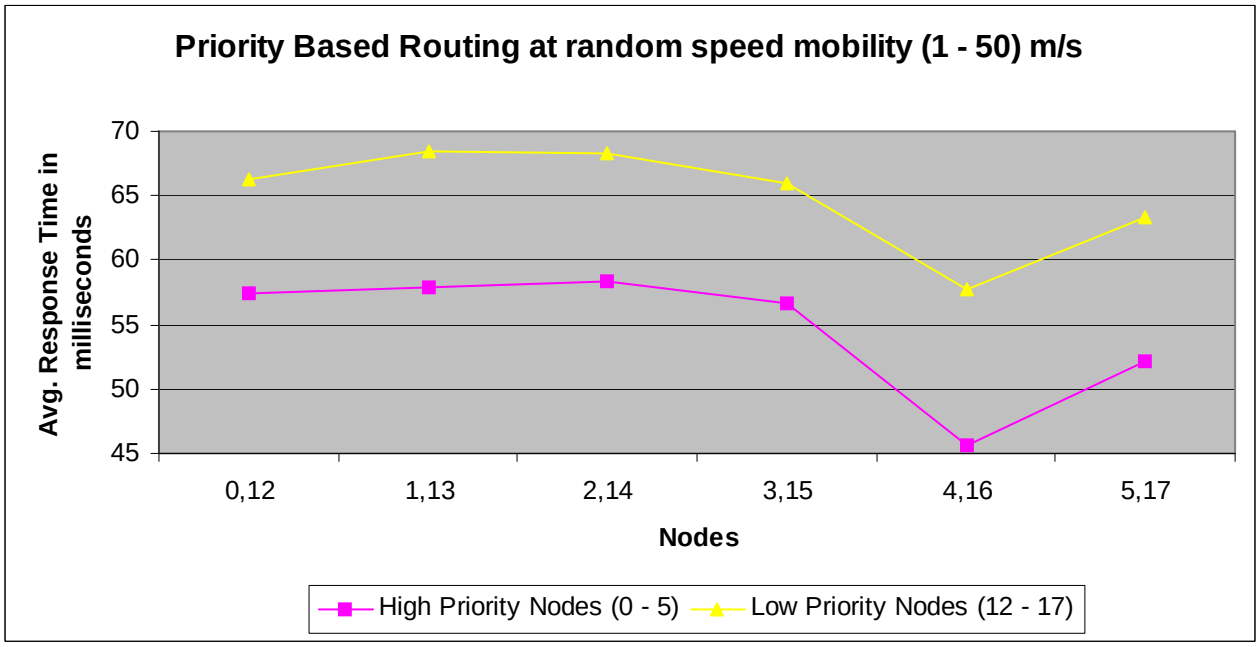

Figure 5-3b: Average response time during priority based routing at random speed mobility

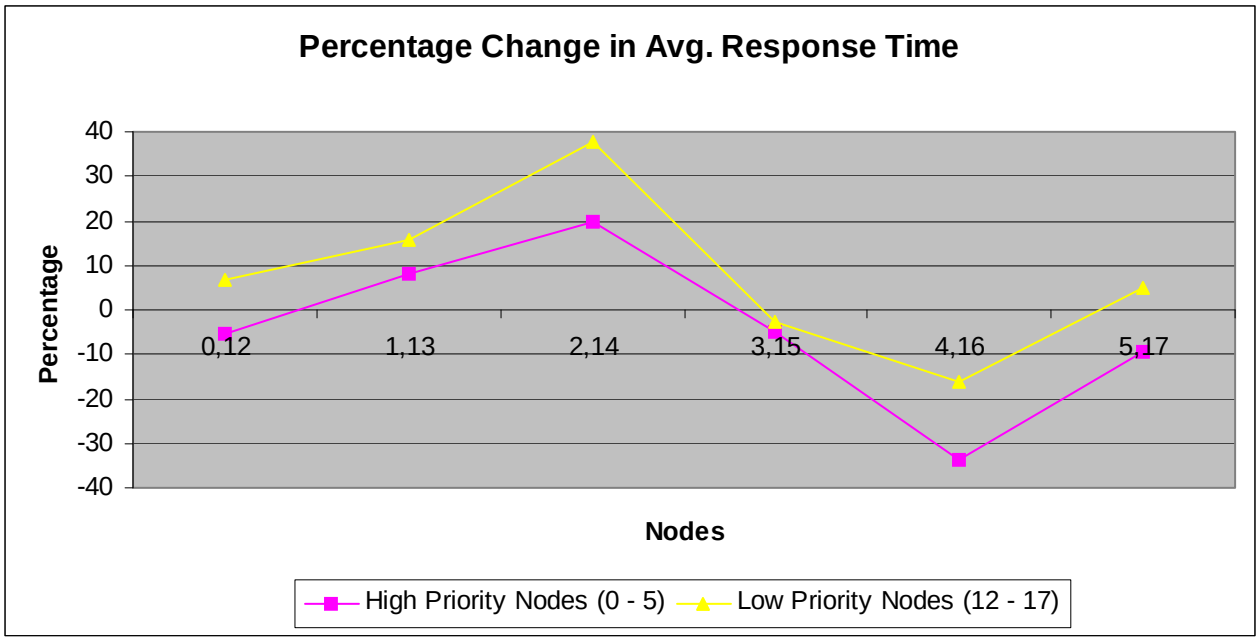

Figure 5-3c: Percentage change in average response time from normal routing to priority based routing at random speed mobility

The average response time of high priority nodes has mostly decreased during low speed, high speed and random speed mobility as shown in Figure 5-4.

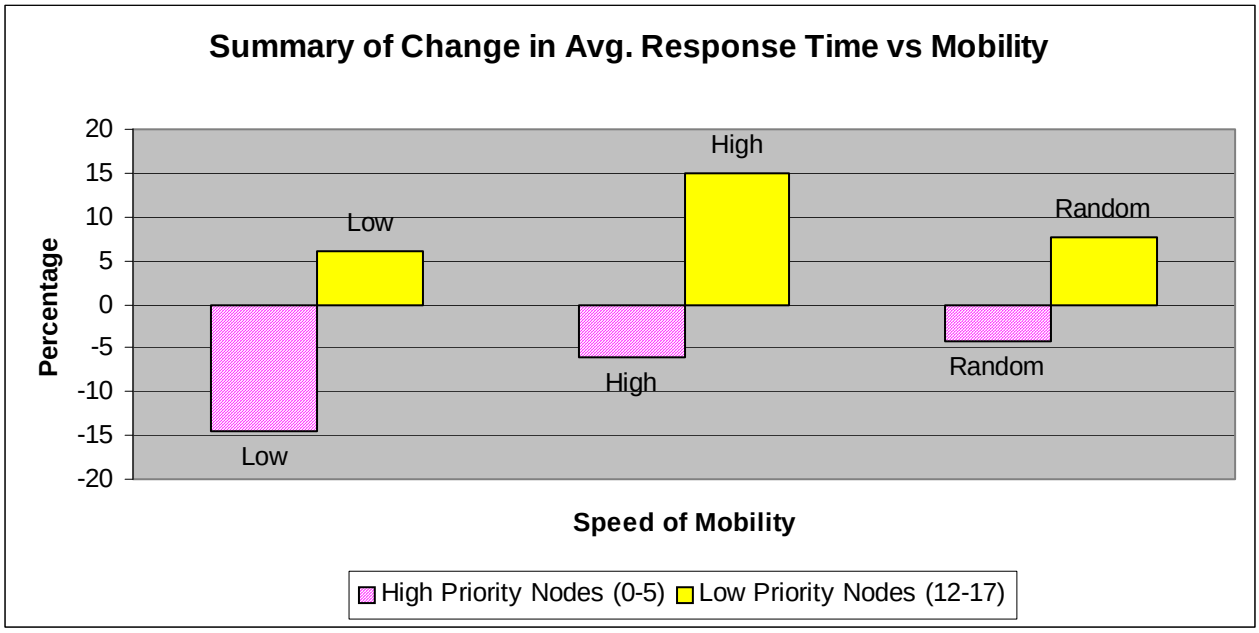

Figure 5-4: Average percentage change in average response time from normal routing to priority based routing vs. Mobility.

5.2.2 Case 2: Nodes have same priority

Traffic is designed such that nodes 0 to 5 and nodes 12 to 17 transmit P2P traffic at the rate of 1.6Mbps for 20 seconds. The traffic pattern is same as shown in Table 5-2 except that now, all nodes have same priority. The average response time of nodes during normal and priority based routing are compared.

Scenario 1: Low speed mobility at 0.5 m/s

All nodes move at constant low speed of 0.5 m/s in random directions. Example: People moving with mobile devices in crowded downtown areas.

The average response time of majority of nodes has increased from normal routing to priority based routing (Figure 5-5a, 5-5b & 5-5c). Hence, this mechanism does not benefit MP2P network when all nodes have same or no priority.

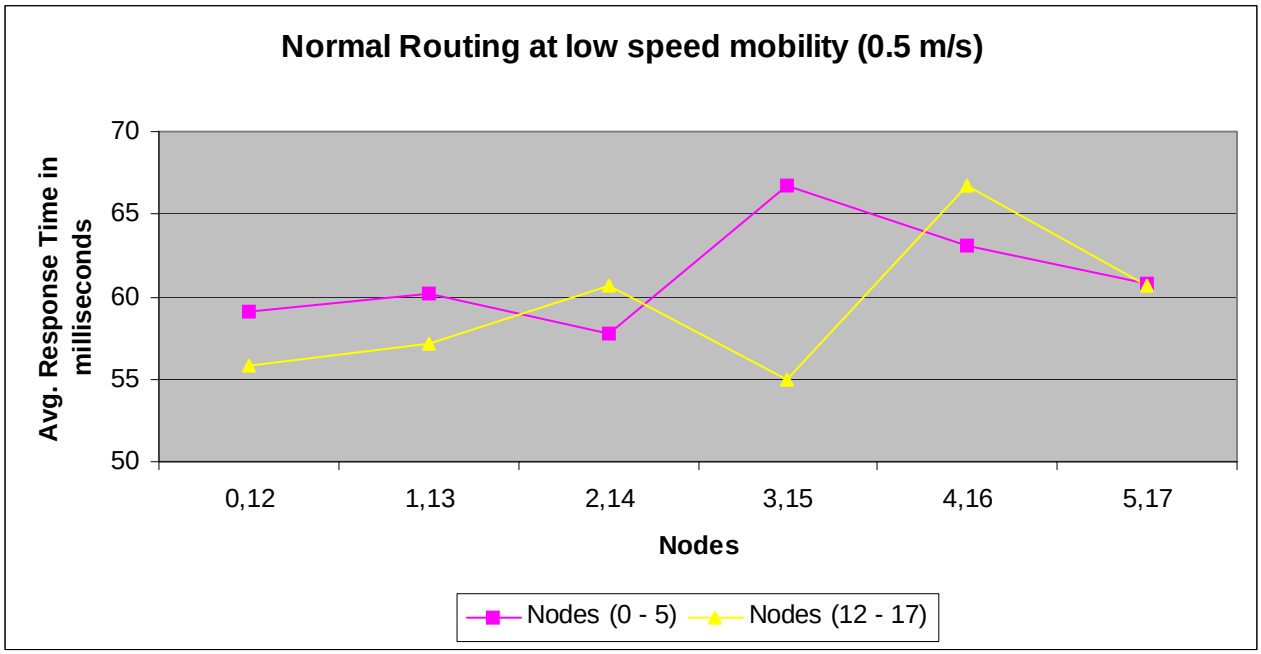

Figure 5-5a: Average response time of nodes during normal routing at low speed mobility.

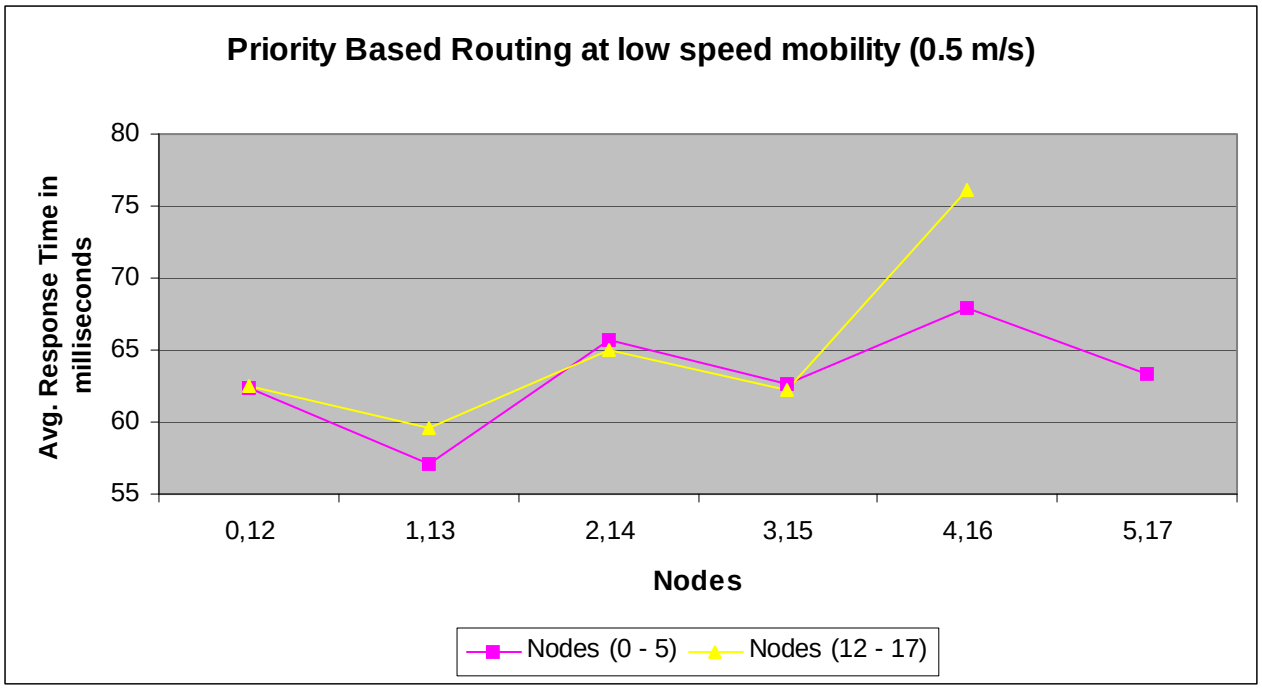

Figure 5-5b: Average response time of nodes during priority based routing at low speed mobility.

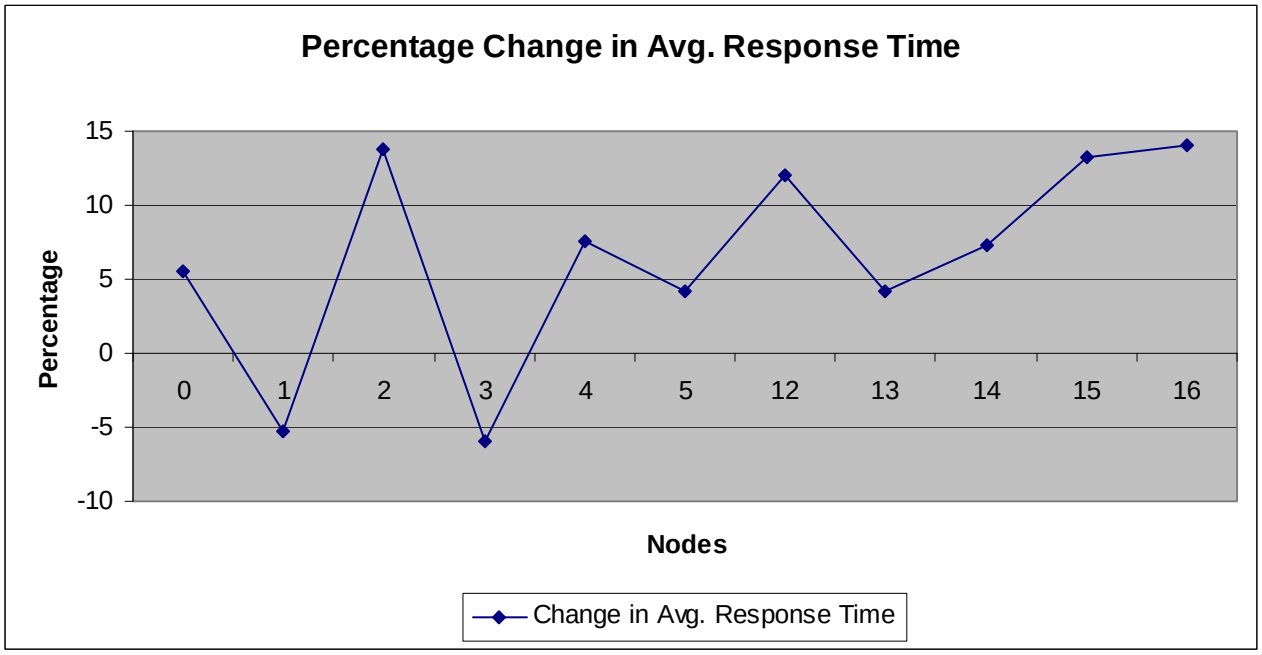

Figure 5-5c: Percentage change in average response time from normal routing to priority based routing at low speed mobility.

Scenario 2: High Speed Mobility at 100 m/s

All nodes follow random direction mobility model where they move at constant high speed of 100 m/s in random directions. Example: People with mobile devices traveling in high-speed vehicles such as buses or trains.

From Figure 5-6a and Figure 5-6b, it is known that average response time of nodes is randomly affected by priority based routing mechanism. The increase in the average response time of few nodes is higher than the decrease in the average response time of few other nodes from normal routing to priority based routing (Figure 5-6c). Hence, this mechanism does not benefit MP2P network when all nodes have same or no priority.

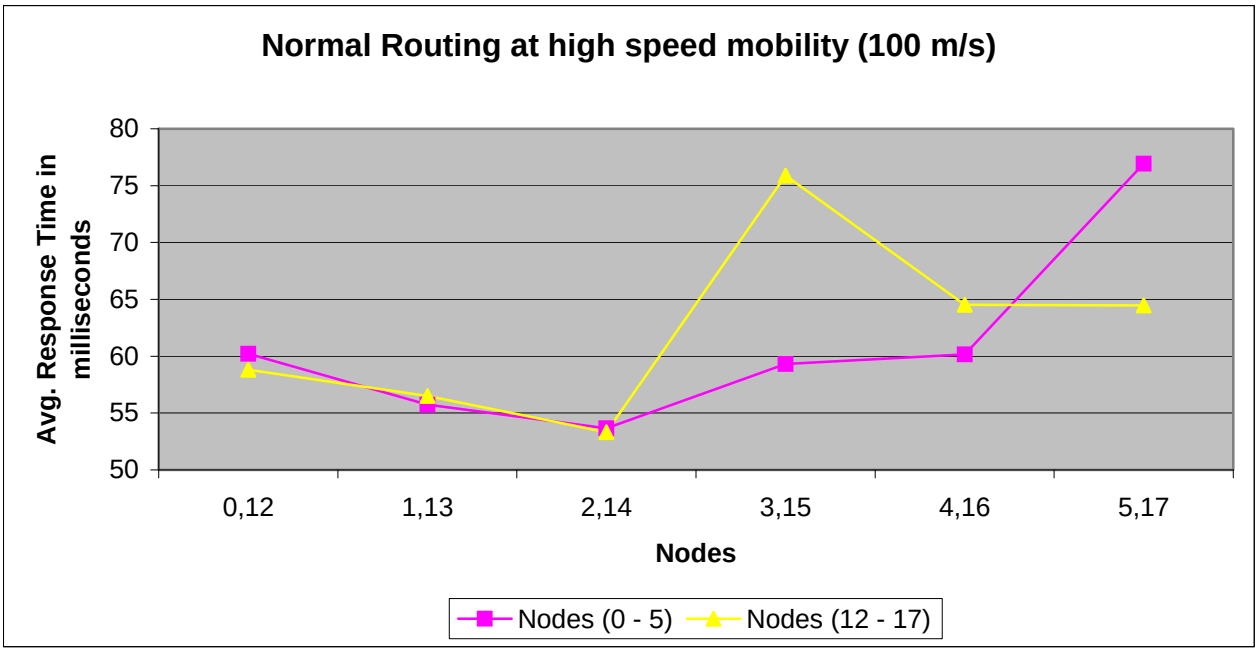

Figure 5-6a: Average response time of nodes during normal routing at high-speed mobility.

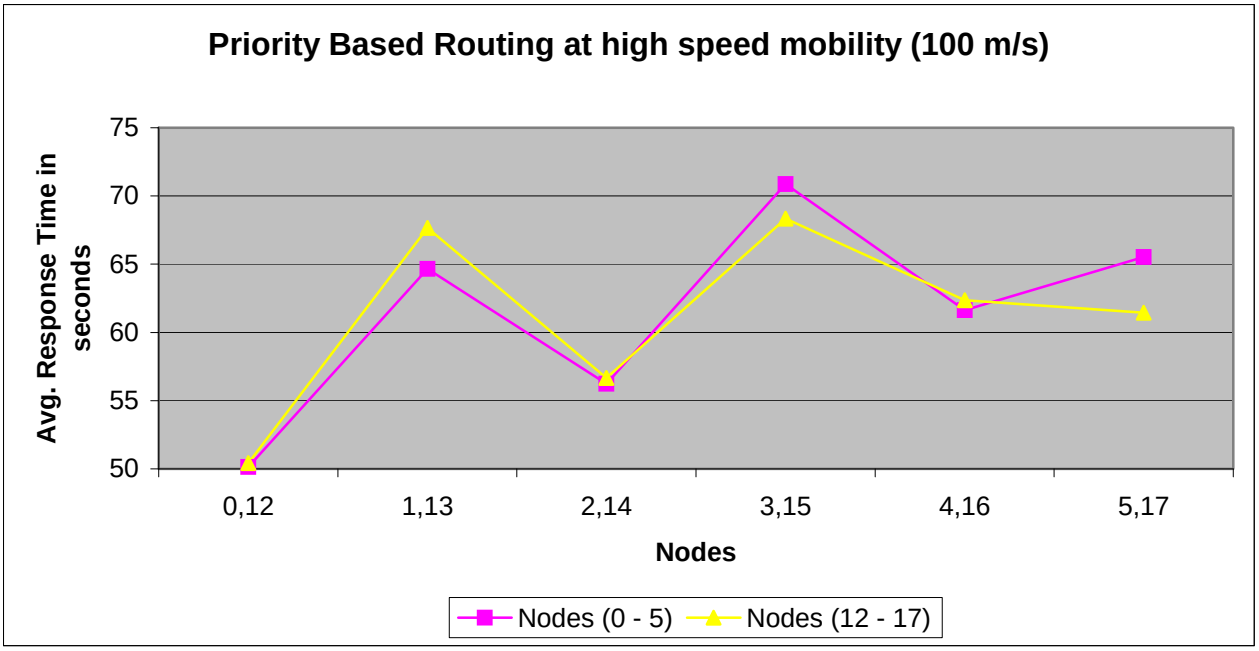

Figure 5-6b: Average response time of nodes during priority based routing at high-speed mobility.

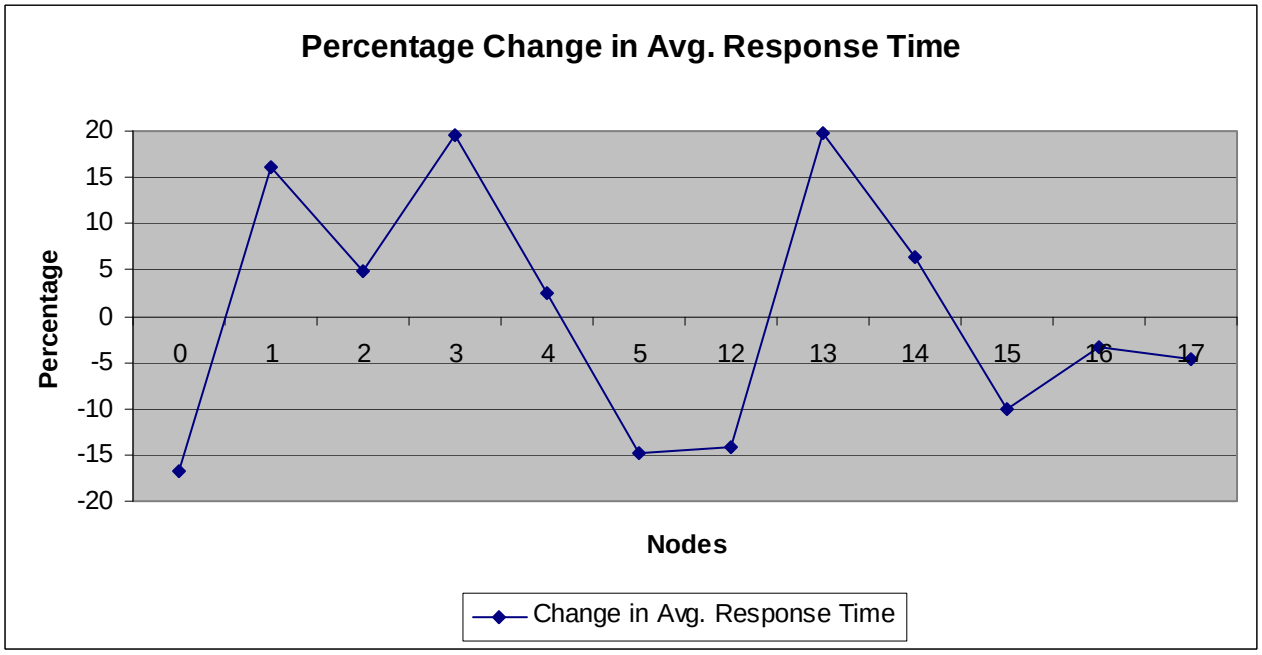

Figure 5-6c: Percentage change in average response time from normal routing to priority based routing at high-speed mobility.

Scenario 3: Random Speed Mobility with speed between 1 m/s and 50 m/s

All nodes follow random direction mobility model where they move at random speed (1m/s – 50 m/s) in random directions.

From Figure 5-7a and Figure 5-7b, it is known that average response time of nodes is randomly affected by priority based routing mechanism. The average response time of majority of nodes has increased from normal routing to priority based routing (Figure 5-7c). Hence, this mechanism does not benefit MP2P network when all nodes have same or no priority.

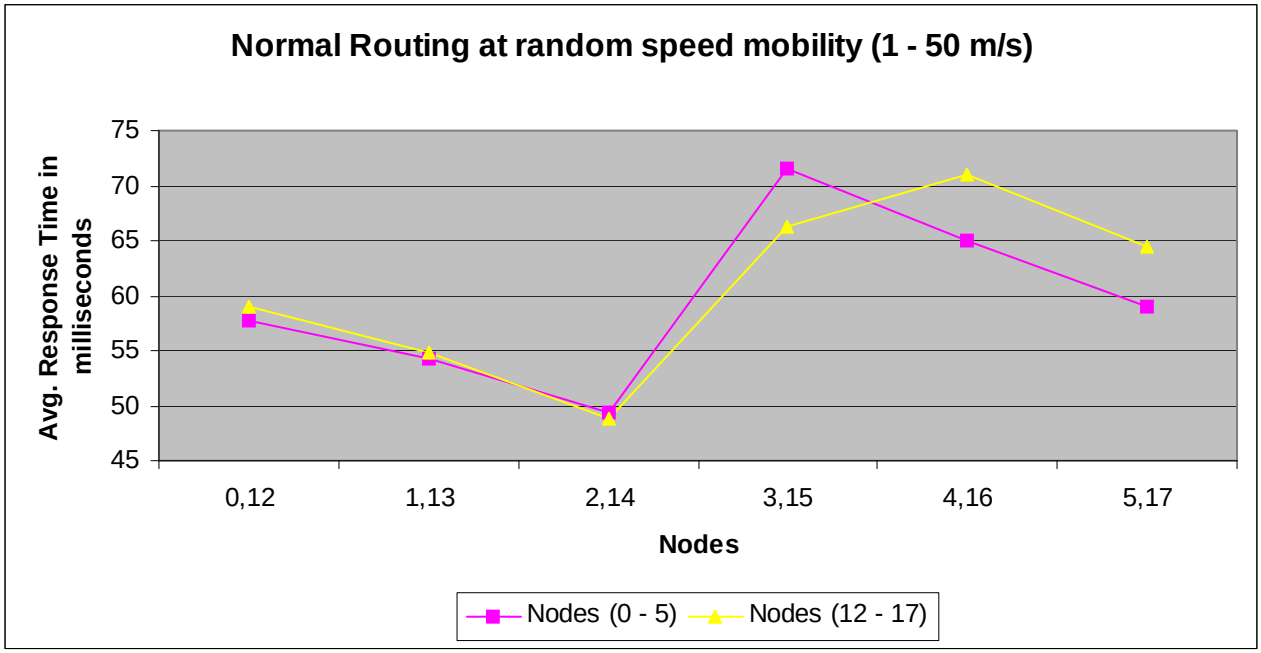

Figure 5-7a: Average response time of nodes during normal routing at random speed mobility.

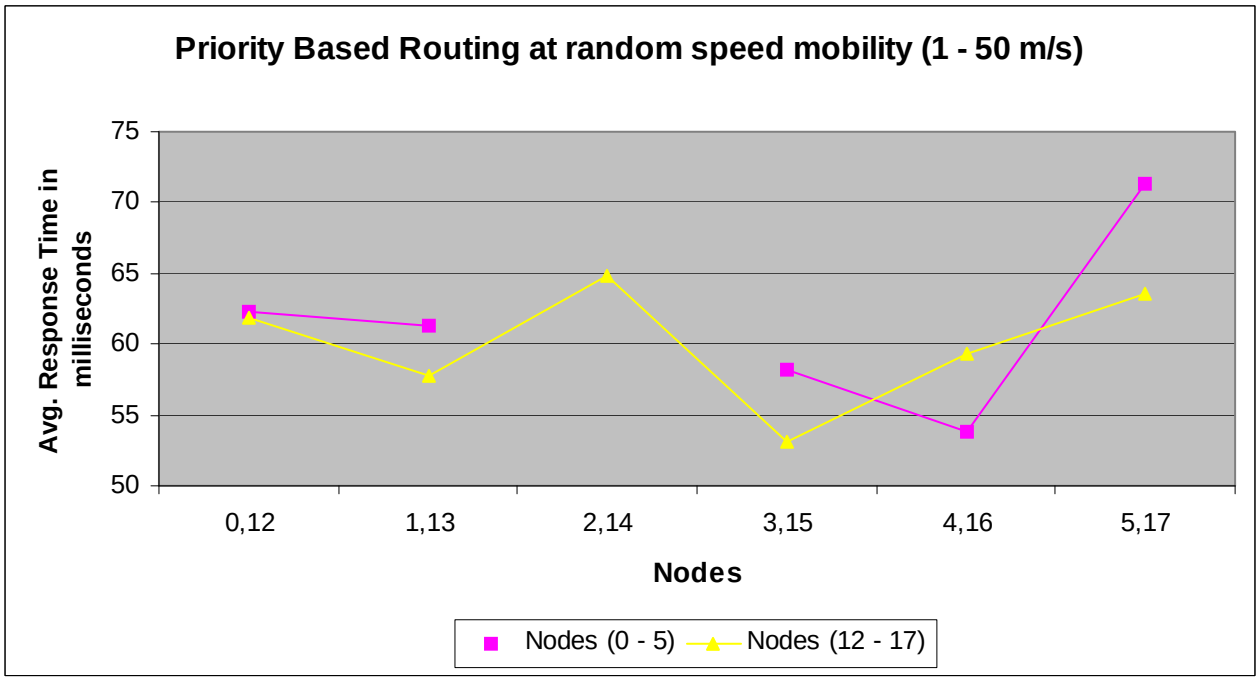

Figure 5-7b: Average response time of nodes during priority based routing at random speed mobility.

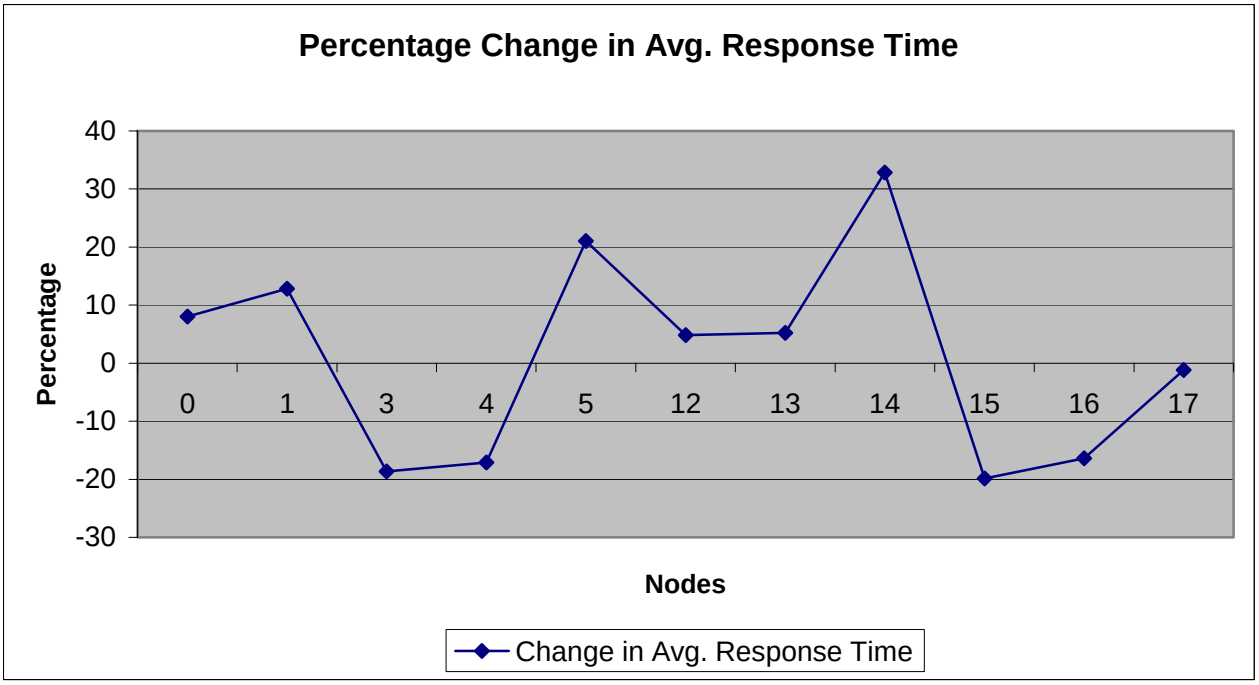

Figure 5-7c: Percentage change in average response time from normal routing to priority based routing at random speed mobility

The average response time of same priority nodes has increased from normal routing to priority based routing during low speed, high speed and random speed mobility as shown in Figure 5-8. Here, high-speed mobility is 100 m/s and random speed mobility is between 1m/s and 50m/s.

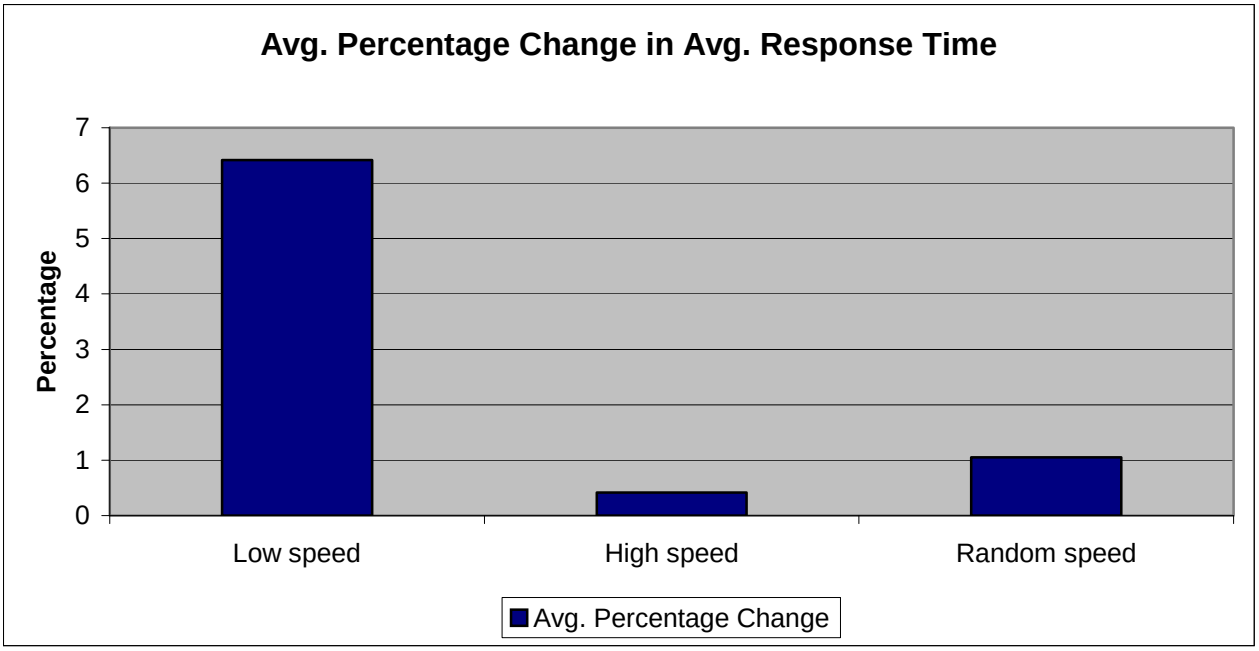

Figure 5-8: Average percentage change in average response time from normal routing to priority based routing.

5.3 Summary

In a MP2P network, where the contribution of nodes varies, the traffic of high contributing nodes is given higher priority compared to traffic of other nodes. This decreases the average response time or traffic delay of high contributing nodes from 1% - 15%. However, in a network where all nodes contribute equally priority based routing mechanism may unnecessarily affect the average response time or traffic delay of nodes.

6. Conclusion

Priority based routing provides higher priority to high contributing nodes and popular services during routing. This reduces the response time needed for those high contributing peers by 1% - 15% compared to response time required in normal routing. This reward would certainly motivate more users to join the MP2P network and contribute more to the network.

The proposed priority based routing mechanism can be improved further by considering additional factors for determining the priority of a peer. For example, a peer that serves more number of service requestors can be given higher priority. This factor may avoid friendly peers serving only

each other and thereby increasing their priority in the MP2P network. The number of service requestors can also be considered for determining the priority of that service. This factor may prevent any single peer from increasing the priority of a service by sending continuous requests for that service. Another example may be using the duration and the amount of network resources a service consumes to determine its priority and the priority of node that offers it. This may prevent single high priority peer or single popular service from monopolizing the network resources. Finally, priority based routing mechanism can be implemented using integrated approach and its performance can be compared with its implementation using layered approach.

7. References

1. Cheng, R., Jin, H., Shi, K., and Cheng, B. (2005). "An anycast-based P2P routing protocol for mobile ad hoc networks". *The First IEEE and IFIP International Conference in Central Asia on Internet*, 5.
2. Gerla, M., Lindemann, C., and Rowstron, A. (2005). "P2P MANET's – New Research Issues". *Proceedings of Dagstuhl Seminar Perspectives Workshop*, 1 – 25.
3. GloMoSim - Global Mobile Information Systems Simulation Library. Retrieved on February 6, 2009 http://pcl.cs.ucla.edu/projects/glomosim/
4. Hofstätter, Q., Zöls, S., Michel, M., Despotovic, Z., and Kellerer, W. (2008). "Chordella – A Hierarchical Peer-to-Peer Overlay Implementation for Heterogeneous, Mobile Environments". *Proceedings of Eighth International Conference on Peer-to-Peer Computing*. 75 – 80.
5. Jacquet, P., Clausen, T., Laouiti, A., Qayyum, A., Viennot, L., and Mulethaler., P. (2001). "Optimized Link State Routing Protocol for ad-hoc networks". *Proceedings of IEEE International Multitopic Conference*, 62 – 68.
6. Kortuem, G. Schneider, J. Preuitt, D. Thompson, T.G.C. Fickas, S. Segall, Z. (2001). "When peer-to-peer comes face-to-face: collaborative peer-to-peer computing in mobile ad-hoc networks". *Proceedings of the First International Conference on Peer-to-Peer Computing*, 75 – 91.
7. Mauthe, A. and Hutchison, D. (2003). "Peer-to-Peer Computing: Systems, Concepts and Characteristics".

8. MobiREAL – A realistic Network Simulator. Retrieved on February 6, 2009 http://www.mobireal.net/index.html
9. ns3: Quick Intro and MANET WG Implementations. Retrieved on March 1, 2009 from www.ietf.org/proceedings/08jul/slides/manet-2.pdf
10. ns-3 Software Architecture. Retrieved on March 1, 2009 from www.nsnam.org/docs/design.pdf
11. ns–3, The Network Simulator: WNS2 Tutorial. (2008). Retrieved on March 1, 2009 from www.wns2.org/docs/wns_tutorial-handout.pdf
12. Opnet Technologies. Retrieved on February 6, 2009 from www.opnet.com/
13. Peng, G., Li, S., Jin, H., and Ma, T. (2004). "M-CAN: a Lookup Protocol for Mobile Peer-to-Peer Environment". *Proceedings of the 7th International Symposium on Parallel Architectures, Algorithms and Networks*, 544 – 549.
14. Persson, S. (2007). Mobile Peer-to-Peer Applications in Cellular Networks. Retrieved August 29, 2008, from http://epubl.ltu.se/1402-1617/2007/257/LTUEX-07257-SE.pdf
15. Rowstron, A. and Druschel, P. (2001). "Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems". *Proceedings of the 18th IFIP/ACM International Conference on Distributed Systems Platforms*.
16. Stoicay, I., Morrisz, R., Liben-Nowellz, D., Kargerz, D., R., Kaashoekz, M.F., Dabekz, F., and Balakrishnan, H. (2003). "Chord: a scalable peer-to-peer lookup protocol for internet applications". *IEEE/ACM Transactions on Networking*. 11, 17 – 32.
17. The NS-2 Network Simulator. Retrieved on February 6, 2009 from www.isi.edu/nsnam/ns/
18. The NS-3 Network Simulator. Retrieved on February 6, 2009 from http://www.nsnam.org/
19. Zahn, T. and Schiller, J. (2005). "MADPastry: A DHT Substrate for Practicably Sized MANETs". *Proceedings of the Fifth Workshop on Applications and Services in Wireless Networks*.
20. Zoels, S., Despotovic, Z., and Kellerer, W. (2007). "Load Balancing in a Hierarchical DHT-based P2P System". *Proceedings of International Conference on Collaborative Computing*.

8. Appendix

Few network simulators like NS-2 [17], NS-3 [9, 10, 11, 18], GloMoSim [3], Opnet [12] and MobiREAL [8] were considered in order to analyze the feasibility of simulating priority based routing. Table 8-1 summarizes the result of this analysis.

	NS-2 [17]	NS-3 [9, 10, 11, 18]	GloMoSim [3]	Opnet Modeler Wireless Suite [12]	MobiREAL [18]
Support for node mobility	Yes	Yes	Yes	Yes	Yes
Support for MANET routing protocol	Yes (Ex: AODV, DSR)	Yes (Ex: OLSR)	Yes (Ex: AODV, DSR)	OSPF	
Support for IP layer transmission	Unicast, broadcast and multicast	No Unicast, broadcast and multicast			
Approach for developing TCP/IP model	Object Oriented approach	Object Oriented approach	Layered approach		
Programming language Used	C++	C++ and Python	Parsec (C based parallel language)	C++	C++
Source code availability for academic research	Yes	Yes	Yes	No	On Request
Documentation availability	Yes	Yes			

User Manuals and Tutorials	Yes	Yes	Yes		
Strengths	Stable simulator supporting many protocols at different layers of TCP/IP model Supported by visualization tools (Ex: NAM)	Standard trace file formats which can be studied by WireShark Better documentation, manual and tutorial support compared to ns-2	Standard APIs between layers to support integrating different protocols in different layers		Considers obstacles, pedestrian and automobile movements in urban areas. Suitable for simulating VANET applications
Weaknesses	Simulator is not well-documented Hard to extend or scale Not well-explained manuals and tutorials	Visualization tools are still under development (Ex: iNSpect)			Not designed for general network applications
Suitable for priority based routing	Yes	Yes	Yes		No

Table 8-1: Analysis of network simulators