IJICT International Journal of Information & Communication Technology

ITRC

Volume 1- Number 3- July 2009 (pp.1-13)

A Backup Routing Scheme for Mobile Ad-Hoc Networks

Amir Esmailpour
Department of Computing and Information Science
University of Guelph
Guelph, Canada N1G 2W1
aesmailp@uoguelph.ca

Nidal Nasser
Department of Computing and Information Science
University of Guelph
Guelph, Canada N1G 2W1
nnasser@uoguelph.ca

Received: September 30, 2008- Accepted: May 7, 2009

Abstract- **In recent years, performance of the Mobile Ad-Hoc Network (MANET) has become an important research area in the wireless networks community. MANET consists of clusters of Mobile Nodes (MNs) on the access network that could be connected to other clusters through a fixed backbone of routers. The access link contention can severely constrain the end-to-end throughput of the path between a pair of source and destination MNs connected through the backbone. In this paper, we propose an integrated routing system for MANET that includes the backbone paths and the ad-hoc paths formed as a result of direct communication among MNs without going through the backbone. In our proposed routing system, an alternative ad-hoc path can be used only when the primary backbone path is severely constrained due to access links contention. We also propose a scheme for making the MN aware of link quality measures, and incorporate throughput metric in the core of AODV. We implemented the proposed routing system in the OPNET simulator, and evaluated the performance of our scheme under a variety of conditions. Simulation results show that the alternative ad-hoc path is effective in delivering higher throughput when the backbone path is severely constrained.**

Keywords: MANET, Wireless Mesh Network (WMN), On-demand routing, Link-state routing.

چکیده - در سالهای اخیر کارائی شبکه های سیار Ad-Hoc(MANET) موضوع مهم تحقیقاتی در حوزه شبکه های بی سیم شده است.MANET شامل خوشه هایی از گره های سیار در شبکه دسترسی می باشد که می تواند بوسیله مسیریاب های ثابت ستون فقراتی به خوشه های دیگر متصل شود. در این مقاله یک مسیریابی مجتمع برای MANET ارائه می گردد که شامل مسیرهای ستون فقراتی ثابت و Ad-Hoc می باشد . در طرح پیشنهادی زمانیکه مسیر ثابت زیرساخت شبکه ناشی از ازدحام موجود محدود شود از یک مسیر دیگر در میان شبکه Ad-Hoc استفاده می گردد و روشی برای آگاهی گره های سیار از میزان کیفیت مسیرها ارائه می گردد. جهت شبیه سازی از شبیه ساز OPNET استفاده شده است و عملکرد روش پیشنهادی در شرایط مختلف ارزیابی می شود. نتایج شبیه سازی نشان می دهد در شرایطی که مسیر ثابت زیرساخت شبکه دارای محدودیتهای فراوان است تاثیر بسزایی در بالا بردن کارآیی شبکه دارد

I. Introduction

MANET and WMN have become popular wireless technologies in the recent years for the practical applications they could bring to some of long-awaited demands of wireless technologies, such as low cost deployment in the areas of poor network infrastructure and terrain of difficult deployment. Wireless routers deployed at fixed locations and connected through wireless links form the backbone in a mesh topology called WMN backbone [1]. User devices are connected to the network through access links. Typically every router in the backbone has two wireless links: a backbone link connected to other wireless routers, and an access link connected to the user devices. Routing in WMNs is a challenging problem because of dealing with the unpredictable behavior of multi-hop wireless links caused by interference, noise, fading, channel propagation, and other MANET characteristics. Most of the routing proposals for WMN use some form of ad-hoc routing with more innovative metrics to reflect wireless link conditions, e.g. Expected Transmission Count (ETX) [2], Expected Transmission Time (ETT) [3], and Weighted Cumulative ETT (WCETT) [3]. In this paper we propose an integrated routing system for MANET that utilizes both backbone and access links to achieve the benefits of multi-path routing. The WMN backbone is formed by fixed wireless routers; hence, its topology does not change frequently. This allows us to employ link-state routing in the backbone such as OPEN Shortest Path First (OSPF). User devices that are connected to different wireless routers in the backbone can also be connected to each other through their access links to form an ad-hoc network. Since user devices are not fixed, but rather move on their own, the topology of the ad-hoc network undergoes more frequent changes. Hence, we propose on-demand routing in the ad-hoc network. Thus, the proposed integrated routing system for MANET provides a source with at least two alternative paths: one through the ad-hoc access network called the ad-hoc path (ah_path) and the other through the backbone network called the backbone path (bb_path). Since wireless routers in the backbone are fixed and connected to a permanent source of power, the backbone paths are relatively more stable with no power constraints. In contrast the access paths are relatively less stable with power constraints due to the mobility and limited power source of user devices. Hence, we suggest that bb_path is used as a primary path. There are at least three situations when ah_path has benefit over the corresponding bb_path. First, the high contention on the access link between the MN and the Access Mesh Router (AMR) can significantly reduce throughput of the bb_path. Secondly, when the excessive handover delay (after a MN moves from one Wireless Mesh Routers (WMR) to another), causes transient bb_path outage. This can be mitigated by setting up an alternative ah_path. Third, the ah_path will be used when it has fewer hops than the corresponding bb_path. This typically happens when both the source and the destination MNs are in adjacent clusters. For instance, for two MNs in adjacent clusters, in close proximity to each other and having direct communication capability, the single hop ah_path may become superior to the corresponding bb_path. The MN will have two paths to choose from: bb_path versus ah_path. In order for the MN to make a decision about which path to choose, we include link quality metrics in the routing protocol used by the MN.

Throughput analysis is provided based on a system throughput that includes throughput estimations for both backbone and ad-hoc networks. We use two different methods to estimate throughput of the two paths using Bianchi's [24] throughput measurements and throughput measurements in [25] respectively.

Our integrated routing system also considers two types of MNs. MNs with one physical interface and two physical interfaces. In case of MNs with two interfaces, using two different radio frequencies, the MNs will reduce channel contention and improve traffic throughput. MNs use one interface to connect to the backbone WMR, and use the other interface to connect to other MNs in the ad-hoc network. MNs with two interfaces, where a user could connect to two different networks using a single device, have become more popular in recent years.

For the access network, we select Ad-hoc On-demand Distance Vector (AODV) as the main routing protocol for the MN and integrate throughput into the core of AODV: in the routing cache and Route Reply (RREP) packet. We implemented this solution in OPNET modeler 11.5 [28], and showed by simulation that AODV performance always improves by providing throughput information to regular AODV. AODV is selected mainly due to the fact that it works on an on-demand basis. We believe that the ad-hoc path will not be used under normal circumstances. It will only be used when the backbone is constrained. Therefore, this algorithm will only be used on-demand as well. Using other ad-hoc protocols could increase overhead and unnecessary traffic.

For the backbone network, we use OSPF routing protocol. OSPF is a proactive protocol suited to stable networks such as the fixed backbone of WMNs. Use of OSPF in the WMN is well defined in literature, and employed by Nortel Networks [30].

The rest of this paper is organized as follows: In Section 2, we present the architecture of MANET in a mesh environment, which is used in this paper. In Section 3, we provide related work in the area of routing for MANET and WMN. In Section 4, we establish design principles used in this work. In Section 5, we evaluate the performance of our proposed routing system, and show the simulation results. Finally, in section 6, the conclusion and future work is presented.

II. MANET Architecture in Mesh Environment

In this section, we discuss architecture of the MANET used in this paper. We also discuss global connectivity and addressing in the backbone and access networks, and how new metrics could help in decision making of MN when it has to choose between two paths: one through the backbone, the other through the access networks.

A. *Backbone and Access Network Structure*

The architecture of MANET we consider in this work consists of a backbone network of WMRs and clusters of ad-hoc networks that are connected through the backbone, as shown in Figure 1. Each MN is connected through an access link to a WMR, which serves as a gateway to the backbone network. Some WMRs in the backbone are connected to the Internet and serve as gateways to the Internet for the entire wireless mesh network.

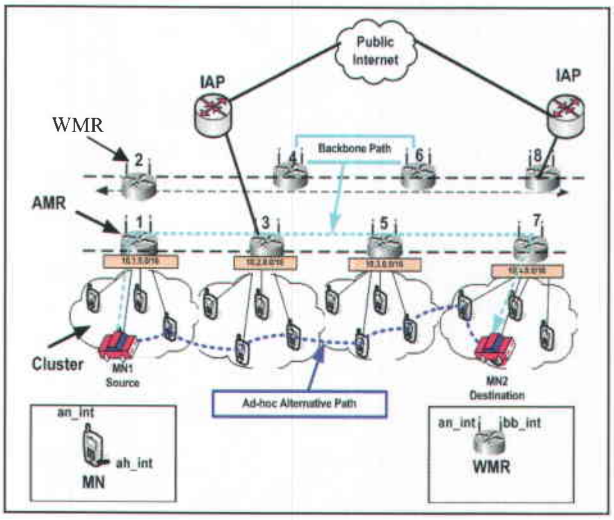

Figure 1: Architecture of the proposed MANET in a mesh environment

The access network in MANET is essentially a group of MN clusters connected via the backbone. Throughout this paper we often use "access network" to refer to "MANET". Each AMR has two 802.11 interfaces: the backbone interface (bb_int) and the access network interface (an_int). Using MNs with only one interface poses several problems such as interference issues, as investigated in one of our previous studies [29].

We use different radios for the bb_int and an_int to eliminate interference between the two paths. All bb_ints are equipped with 802.11a radios and connect WMRs to the backbone, whereas an_ints use 802.11b/g and connect WMRs to the MNs in the access network. The bb_int is configured in 802.11 infrastructure mode of operation, whereas the an_int is configured in 802.11 ad-hoc mode (Figure 2).

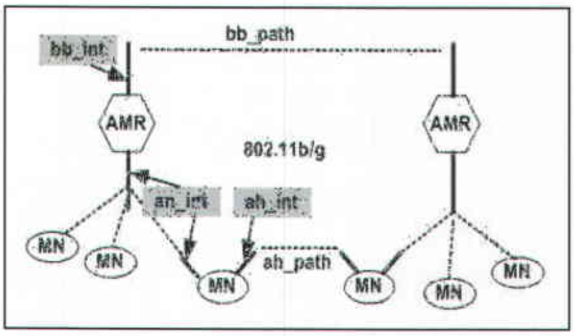

Figure 2: Both AMR and MN have two interfaces; backbone and access, using 802.11a and 802.11b/g respectively

Each MN is equipped with two interfaces, an access network interface, called an_int; and an ad-hoc interface, called ah_int. The MN is connected to the backbone via a WMR through an_int. It uses ah_int to form the ad-hoc network of MNs, discussed in Section 4. Both interfaces can be implemented using 802.11b radios configured in ad-hoc mode on different channels. The ah_int of all the MNs in the network are configured on a single channel to form the ad-hoc network.

B. *Global Connectivity and Addressing*

The entire MANET is an IP network where some WMRs are connected to the Internet, called Internet Access Points (IAP). A WMR may be connected by several virtual links to multiple mesh routers through its bb_int. We assign a different IP subnet address to each of those links. Hence, we create as many sub-interfaces (virtual interfaces) on a bb_int as the number of subnets the WMR is connected to. The an_int forms the access link, which is assigned an IP subnet address. Thus, all the MNs that are connected with a WMR through its access link receive an IP address on that subnet. All the MNs connected with the same WMR form a cluster, where the WMR becomes the clusterhead, called the Access Mesh Router (AMR). When a MN approaches the vicinity of a WMR, it receives the an_int beacon and connects with the WMR. If it moves from the coverage area of an_int of the old WMR to the new WMR, then it performs handover and changes its IP address by acquiring a new address on the subnet of the an_int of the new WMR. We allow the connectivity between a MN and its AMR through a multi-hop path composed of mobile nodes within the same cluster. Hence, a cluster of MNs and the associated AMR form an ad-hoc network. The mobility at IP level can be managed by employing a variation of the IP mobility solution discussed in [10]. We do not further discuss mobility and power control managements in the proposed solution since they are out of the scope of this paper.

C. *Routing in the MANET Architecture*

We propose an integrated routing system for MANET that considers characteristics of both backbone and access networks. Between a pair of source (S_MN) and destination (D_MN) MNs, there are 2 paths: ah_path and mesh_path. For the ah_path we use AODV multihop going through the ad hoc network (Figure 3). The mesh_path has three components; sub-path1 between S_MN and S_AMR, sub-path2 between S_AMR and D_AMR (also called bb_path) and finally sub-path3 between D_AMR and D_MN. Sub-path1 and sub-path3, although part of the mesh_path, are in fact access links and use AODV to establish that link.

We modify the AODV routing protocol such that AMRs acting as clusterhead periodically send beacons to discover neighbors (all the MNs in their respective clusters), and keep their local cluster's MNs in their AODV cache tables (or IP forwarding table). Thus, when an AMR receives a packet from another AMR, it will find the subnet and forwards the packets to the corresponding AMR, which in turn forward the packet to the D_MN.

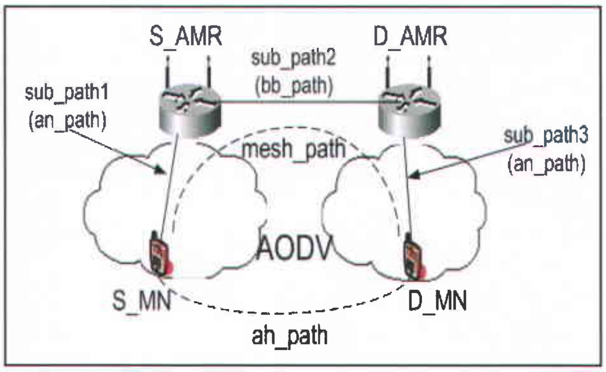

Figure 3: Access network, both an_path and ah_path use AODV

On the access side, MNs use AODV, so if the D_MN is located in the same cluster as S_MN, then the route is discovered and packets are sent directly to the D_MN, with AMR not intervening at all. When the AMR receives a RREQ (Route Request) in which D_MN is in the same subnet as the S_AMR, the packet is dropped, assuming that there is a direct connection between the two MNs in the same cluster.

III. Literature Review

In this section, we review related research from the literature in different areas of routing for both MANET and WMN, as well as some approaches by the research community for enhancing routing performance, such as MeshDV [9].

A. *Routing Protocols of MANET and WMN*

Conventional routing protocols designed for wired networks could not satisfy the unique characteristics of ad-hoc networks, leading to the design of new routing protocols exclusively for ad-hoc networks. MANET is characterized by mobility of nodes, limited power supply, and unstable routes. These characteristics of MANET introduce continues topology changes that would create an enormous amount of overhead and flooding, and excessive number of calculations, if conventional protocols were used. Several new routing protocols have been proposed to overcome many shortfalls of the traditional protocols when used for ad-hoc networks.

A vast amount of research has focused on routing of ah-hoc networks in the past decade. Several surveys are available covering and summarizing numerous publications in different areas of routing for ad-hoc networks: [13, 14, 15] just to name a few. Numerous routing protocols propose hierarchical routing, cross-layer designs, clustering and so on. One of the most common ways to characterize those routing protocols is to divide them into reactive and proactive groups. Proactive protocols, such as Highly Dynamic Destination-Sequenced Distance Vector (DSDV) [16] and Optimized Link State Routing Protocol (OLSR) [17] keep routes in their routing table, and periodically update them. Reactive protocols, such as AODV [18] and Dynamic Source Routing (DSR) [11, 19], on the other hand, work on a need-driven basis, where a route discovery is only initiated based on-demand.

Wireless medium characteristics affect behaviors of wireless networks in many ways. Such characteristics include channel fading, medium access contention, and interference, as well as other physical and MAC layer issues. To this effect, routing protocols for wireless networks, albeit at the network layer, should be able to address such lower layer problems. This point leads to the idea of cross-layer design for routing protocols where the lower layer characteristics could be informed to the network layer in forms of new metrics that could be incorporated into the layer-3 packet header. Numerous studies have proposed routing solutions for WMNs [7, 12, 21, and 26]. Reviews of cross-layer designs and proposed metrics are presented in [20] and [21]. Iannone [7] introduces new metrics for interference and packet success estimation ratios that are communicated among Physical, MAC and Network layers.

To add even more complexity to the wireless medium, MANET is characterized by high mobility and low power supply. These features also depend on physical and MAC layer characteristics. In several studies, researchers have shown that traditional routing metrics, such as hop-count, are not suitable for ad-hoc networks. Introduced by De Couto et al. at MIT, the idea of "Shortest Path is not Enough" [12] has become a new paradigm attracting many researchers to introduce several new metrics for ad-hoc routing protocols. They believe any new metrics for MANET or mesh routing protocols should carry link quality or physical layer information.

Designing appropriate metrics has major impact on the backbone routing. ETX measures the Expected Transmission counts of successful packet deliveries as defined in [2], and is effectively used in selecting high throughput paths. ETX is rendered ineffective if WMRs are configured with multiple interfaces, as shown in [5]. Since ETX finds links with low loss rates, in many cases it ignores high bandwidth paths. For example, ETX tends to choose 802.11b as it shows a lower loss rate than 802.11a, even though it provides much less bandwidth. Hence, new metrics such as ETT and WCETT are proposed in [7], which measure ETT and WCETT. These metrics can be used to find paths with higher throughput and lower interference.

On the same note, Drave et al. compared link quality metrics with traditional hop-count [5]. They also concluded that in ad-hoc networks where mobility is high, topology keeps changing. For this reason, any kind of link quality metrics (ETX, ETT, WCETT) would not perform well, since every time topology changes, they should recalculate the link quality metrics with the new topology. All these repetitive calculations will introduce a large amount of delay and reduce throughput. All those studies conclude that the ad-hoc nodes react quickly to fast topology changes; therefore, hop-count would perform better than link quality metrics in ad-hoc networks.

Link quality metrics are computed in the network layer by measuring packet counts [27]. Another approach is to use cross-layer design to compute a metric. In [4] new metrics for interference and packet success estimation ratios are proposed that are communicated across Physical, MAC and Network layers. There are also other studies that show that Quality of Service (QoS) parameters could also be incorporated in the routing by using QoS metrics [22].

In our design, we look at link quality metrics for two types of paths between a pair of source and destination MNs. bb_path and ah_path could show different link qualities with respect to each other. Such differences between characteristics of bb_path and ah_path indicate that in designing a routing protocol that embraces both paths, we should include characteristics of both paths respectively, and consider separate metrics for each path.

B. *Integrated Routing Protocol for WMN*

WMNs were introduced to overcome shortfalls of ad-hoc networks such as power shortage and mobility issues of ad-hoc nodes. Most important of such shortfalls occur in the routing. Ad-hoc networks could not use traditional routing protocols mainly due to ad-hoc characteristics such as mobility and power constraints [5]. However WMNs do not suffer from those constraints. WMNs are characterized by fixed WMRs in the backbone that have unlimited power supply. So, theoretically traditional protocols with some modifications and improvements could be used again [26]. New solutions involving these ideas usually ignore ad-hoc constraints and try to improve routing performance in the backbone by introducing new metrics to the original protocols [3, 5, 7, and 12].

Research in routing for the backbone has mainly concentrated around new metrics that promise performance improvements. However, one should not ignore the fact that WMN is not only in the backbone. WMN includes a major section on the access side, which still falls into the ad-hoc networks and carries all the characteristics of ad-hoc. In order to address routing in WMNs, we must clearly distinguish the characteristics of backbone and access, and realize the fundamental differences between the two distinct parts of the network. WMN is comprised of fixed backbone, and mobile ad-hoc access sides. An integrated routing protocol that could address the needs of both networks should be aware of the path characteristics of each network and treat each network according to its own characteristics [9].

To our knowledge, only a few researches have looked at integrated routing through backbone and access in WMNs. Iannone et al. proposed MeshDV [9, 22], which is a comprehensive routing system that takes into consideration both the backbone and the access sides of WMN. MeshDV combines proactive routing for the backbone with a reactive component for the client side. In MeshDV architecture, there is a client manager module that keeps two tables: a Local Client Table (LCTable) and a Foreign Client Table (FCTable). The LCTable holds information on all the clients associated with a WMR, similar to MNs in our clusters, and a list of all WMRs that have inquired about these MNs. The FCTable holds information on all non-local clients, and a pointer to their corresponding WMR. In the solution by Iannone et al, WMRs perform all the work, and hold all the information. Mobile nodes are not involved in routing decisions. The backbone is transparent to the mobile node. Like MeshDV, we also consider both backbone and ad-hoc access for routing. However, in our solution the routing and decision-making are distributed between WMRs and MNs. We use a route table instead of a FCTable, and do not need to keep routes from non-local clusters in the route table of each WMR. We also use a regular AODV cache table instead of an LCTable

On another note, most proposed WMN routing solutions to date, improve performance based on link quality solutions to overcome link failure [2, 8, 14, and 26]. However they do not address node related issues such as node failure, Denial of Service (DoS) and cluster-head hotspot congestion. Although rare, a node failure could potentially disconnect the corresponding cluster from the network. Therefore, we believe a comprehensive routing solution should address node failure issues as well. Our proposed solution will also address node related issues by providing an ah_path which is completely independent of the WMRs and the backbone, and could be used as a backup to the mesh_path, should a WMR fail or become totally unreachable due to DoS.

C. *Cross-Layer Design for MANET*

Another approach is to use cross-layer design to compute a metric, or to share link quality information among layers. In [9], new metrics for interference and packet success estimation ratios are proposed that are communicated across Physical, MAC and Network layers. Reviews of cross-layer designs and proposed metrics are presented in [6,7]. There are other studies that show that QoS parameters could also be incorporated in the routing by sharing QoS metrics in a cross-layer design [8]. Our proposal, although not a complete cross-layer design solution, helps routing by providing throughput information to the network layer for the decision-making process.

IV. Integrated Routing System Design

In this section, we explain design of the proposed integrated routing system for MANET. We discuss routing for the end-to-end path between a pair of source and destination MNs. In the backbone several routing protocols have been proposed such as AODV with different extensions, DSR, OSPF and so on. We designed OSPF in the backbone and AODV for the access and ad-hoc networks as explained in the following sections. OSPF is a proactive and table-driven protocol, whereas AODV is an on-demand protocol. To the best of our knowledge, there has been no implementation for redistribution between these two protocols yet. Therefore, for the purpose of this study we use OSPF and AODV separately for backbone and access networks respectively, and where necessary we have provided routing information through extensions for AODV in the backbone. We study the routing system for MNs with one or two interfaces and allow MN to choose the ad-hoc network over the backbone under constrained condition in both cases. In order to make the decision to choose the ah_path, MN will be informed of throughput information via AODV RREP packet, which we call throughput-aware AODV.

A. MANET Routing Requirements

The access network includes clusters of MANET nodes. Consider paths between S_MN and D_MN in Figure 3. We defined ah_path and mesh_path. Mesh_path includes an_path and bb_path. Throughout this paper, we use the terms mesh_path and bb_path interchangeably when comparing them to ah_path.

Generally, the paths within the backbone (bb_path) are more stable than an_paths because the WMRs are stationary nodes and the links among them are formed by directional antennas; therefore, a proactive routing protocol such as OSPF with the addition of some link-quality metrics is well suited for the bb_path [30]. Dynamic link quality metrics such as ETX and WCETT can be used in the backbone routing to perform multi-path routing within the backbone. However, the an_paths are the unstable segments of the bb_path due to channel contention, rate drops caused by increasing distance between a MN and the AMR, and instability due to node mobility. Hence, the an_paths could constrain the quality of a bb_path, for example, by lowering throughput or raising delay.

We propose bb_path as the primary path used between a pair of source and destination MNs because of its tendency to traverse stable backbone links. We also propose using an ah_path as an alternative when bb_path is severely constrained under the three conditions mentioned previously.

Any ad-hoc routing protocol can be used to establish an_paths. For example, a table-driven proactive protocol can be used because an_paths are part of primary paths that are mostly used. On the other hand, the ah_path is a secondary path that should be set up only when required. Hence, for the ah_path set up we propose using on-demand and ad-hoc routing protocol, such as AODV (Figure 2), which only initiates route discovery if a route is needed. On-demand routing protocols, such as AODV, perform route discovery when a new route is needed for packet forwarding or when an existing route is refreshed in the routing cache. The route discovery process typically involves flooding of discovery packets inside the network, e.g. flooding of ROUTE REQ packets in AODV. Since routes are not discovered or refreshed periodically in on-demand routing, less flooding overhead is incured, which is suitable for MANET on the access side of WMN.

B. Integrated Routing for Access and Backbone

There are three key issues in designing the integrated routing system. First, which node should decide about using either a primary or alternative route? The route selection decision can be made by either the AMR or the MN itself. In either case, the ah_path is established by the MN. Hence, if the AMR makes the decision, then the information about the ah_path has to be transferred to the AMR, which necessitates discovering the full ah_path prior to making the decision. If the MN makes the decision, then it can delay discovering the ah_path after making the decision. The MN can make the route selection in two steps. In the first step it decides about initiating the route discovery based on the quality of the available bb_path. Then, it can decide about using the primary or the alternative path until after the full ah_path discovery and having the knowledge of the quality of the ah_path. Hence, we propose that MN should perform route selection.

Second, when should the MN initiate the route discovery process for the ah_path? The ah_path route discovery is an expensive process; hence we argue that it should be initiated only when there is a good chance of using the ah_path. We propose an algorithm for initiating route discovery in AODV, which is invoked by the MNs. The source MN broadcasts AODV RREQ for the destination setting the AODV RREQ-TTL = x, where x is the number of hops the MN is away from the AMR. When the AMR receives the RREQ from the source node, it checks the destination IP address. If the destination is in the local cluster, the AMR sends regular AODV RREP if it finds the route in its AODV cache. If the destination is not in the local cluster, the AMR propagates the RREQ to the next hop, and sends the RREQ hop-by-hop to the final destination. The D_MN prepares a RREP packet which includes the throughput information as a new field, and forwards the new RREP packet back to the source.

The third important issue in design is how to decide between the quality of the bb_path and ah_path. Dynamic link quality metrics such as ETX and WCETT are effective measures of the throughput of backbone routes [2] and [3]. However, they are not as effective in an ad-hoc network [5]. A careful estimate of round-trip time (RTT) of the ah_path could be used as a measure of ad-hoc throughput. In our analysis, we used throughput as a performance measure. Each node has throughput information of its own link, which could be transferred to other nodes through backhaul transmission via piggybacking with control messages, or creating a special protocol for transmitting the throughput information.

The design of an integrated routing protocol for WMN involves two major components: the first is the Route Discovery process in which MN finds the routes through both mesh_path and ah_path. In this situation the MN evaluates performance of the mesh_path and decides whether to use this path or to discover an alternative path through the ad-hoc network. The second component is path selection which involves evaluating and comparing the route through the mesh_path and ah_path and to decide when the backup path should be used.

1) Route Discovery in Constrained Conditions

S_MN broadcasts an AODV RREQ for D_MN. This RREQ could be captured by either another MN or by a WMR. The MN could be in the local cluster or in a remote cluster. The WMR could be the local clusterhead (AMR) or any other WMR along the way.

Algorithm 1 outlines how the route discovery is implemented. We define a function for initiating the second_route_discovery, which is called-up every time hop count or throughput falls below a threshold value. MN then waits to receive a RREP. Upon receiving RREP, MN checks to see if RREP is from an AMR or another MN. If it is from an AMR, then it should call the second-route-discovery function. This function checks the hop count and throughput of the RREP, and if they fall below threshold, it initiates the second_route_discovery by sending a second RREQ; otherwise, it will enter the RREP into the route table. If the RREP is received from another MN, then it has to check whether the next hop of that MN is an AMR. In either case, the MN still calls the second-route-discovery function. The difference is that if there was an AMR along the way, then the route type would be entered in the route table as backbone.

```
Algorithm 1:
#define function second_rte_discovery
set hc0 = 3;
set Tput0 = 0;
broadcast RREQ;
upon receiving RREP;
  If( route provider ip address == gw ip address)
      Call function second_rte_discovery on ah_int
  else If ( route provider ip address != gw ip address)
          if (NH == AMR)
              Call function second_rte_discovery on ah_int
          else
          for (1 to hc)
              if (rte_type == bb)
                  Call function second_rte_discovery (ah_int)
                  Enter route as bb_path
              elseif (rte_type == ah)
                  Call function second_rte_discovery (ah_int)
                  Enter route as ah_path
              endif
          endif
  endif
end

#function second_rte_discovery
 GET hc and Tput
   if (hc < hc0) | (Tput < Tput0)
          then initiate route discovery via ad-hoc
           & broadcast RREQ (ttl = hc)
   else
           accept the route and inter in the route table
end
```

When a RREQ is received by a MN, first the IP address of D_MN is checked. If the D_MN is in the same subnet, it means that it is in the local cluster. In this case a regular AODV procedure can be used to resolve the route discovery. If the D_MN is not local, but the route to D_MN is available, a RREP is sent to S_MN including the D_MN IP address, its hop count, and the throughput of the route. When an AMR receives the RREQ from S_MN it checks the D_MN IP address. If the D_MN is in the local cluster, AMR uses the AODV cache and replies with a RREP, including the IP address of the destination, just as in regular AODV. If the D_MN is not in the local cluster, the AMR looks up the routing table. If it finds a route to the destination, it returns RREP with the number of hops. A new field called route_type is added to RREP packet. route_type can have values "BB" (for bb_path) or "AH" (for ah_path). RREPs from the backbone are marked as bb_route, whereas RREPs from other MNs are marked as ah_route. Also a new column is added to the AODV route table as "route_type". Any route returned by the mesh router is entered in the route table as bb_route or ah_route depending on where it comes from. Once a RREP is sent by D_MN, it is tagged as ROUTE_TYPE_AH. At any stage, if it passes by an AMR or WMR, its route type will change to ROUTE_TYPE_BB and will remain the same until it reaches the S_MN. Therefore, if a RREP is tagged with AH for its ROUTE_TYPE once it reaches S_MN, that means this route lies entirely within the ad-hoc path, and there is no backbone router on the way of this path.

Upon receiving RREP from AMR, MN has to decide whether the route provided by the AMR satisfies a certain set of threshold requirements. If the required metrics fall below thresholds, then the MN should start a new route discovery by sending a second RREQ using AODV expanding ring search, and finds a secondary route through ah_path and uses it as a backup route.

A set of threshold values for the throughput is defined within the routing information. Upon receiving RREP from AMR, S_MN compares the throughput value collected from the bb_path to the pre-set conditions, and decides whether to use the route provided by the mesh router, or to initiate a new AODV RREQ with longer TTL (Algorithm 1). S_MN initiates route discovery by broadcasting RREQ to ad-hoc nodes, and searches for a backup route via the ad-hoc multi-hop path. Upon receiving RREP from the ad-hoc network, S_MN enters a route in the route table as ad-hoc route_type.

2) Path Selection: bb_path vs. ah_path

Having performed a second route discovery S_MN has two routes to the destination: one through the mesh backbone, and the other through the access ad-hoc network, and has to decide when to use the primary bb_path, and when to switch to ah_path. For this purpose we design Algorithm 2. Algorithm 2 below is a network level implementation of the MN decision-making process.

This algorithm is only executed if there are two routes to choose from. It checks the route table; if there is no route, then it calls Algorithm 1 to find the route. If there are two routes available, and it has to decide which one to take, then it checks the throughput provided by the two routes. The AMR computes the Bianchi's saturation throughput [24] of the two access links and takes the minimum of the two as the mesh_path throughput, and send that to the source MN. For ah_path we suggest using the algorithm of computing the ad-hoc path throughput proposed in [25]. The algorithm calculates the threshold value of "d" by subtracting the two throughputs, dividing them by the ad-hoc throughput and multiplying by 100. This equation will find the difference in the values of throughput for the two paths shown in percentage. Different network set-ups could assume different values for d depending on how reliable the backbone route is. For the purpose of this paper, we assume that we will take ah_path, only if it provides a higher throughput by at least 25%. Theoretically, ah_path could be used whenever the bb_path provides lesser throughput than ah_path. However, in this experiment, a heuristic method is used by trying different threshold values. The result shows that changing the path every time the throughput falls, could generate more calculation and cause a large amount of overhead. Choosing a threshold value of 25%, for the throughput improvement, ensures that the change will take effect only if the ah_path could substantially improve the throughput achieved, or if the backbone is severely diminished.

According to algorithms 1 and 2, the MN uses bb_path until throughput falls below the minimum requirement (threshold). Then once the MN is notified of this information, MN starts a second route discovery (Algorithm 1), finds the ah_path and starts using this path if necessary (Algorithm 2). These algorithms ensure that the MN switches to ah_path whenever throughput falls below the threshold level. Such cases could happen when MN is moving between clusters and there is latency, disconnection, or congestion.

```
Algorithm 2:
route required
check route table
        if no route available
                then start algorithm 1
        else
                check throughput fields of RREQ 1&2
                SET throughput = thr_0
                GET bb_path throughput = Tput_bb
                GET ah_path throughput = Tput_ah
d= (Tput_ah – Tput_bb)/Tput_ah))*100
        if (d > 25)
                then ah_path → status = active
        elseif (d <= 25)
                then bb_path → status = active
        endif
end
```

In our analysis, we used throughput as a performance measure. Each node has throughput information of its own link, which could be transferred to other nodes through backhaul transmission via piggybacking with control messages, or creating a special protocol for transmitting the throughput information. For the purpose of this paper we rely on throughput measurements performed by OPNET. In future we are planning to incorporate link quality metrics for backbone throughput and use more static measurements for ad-hoc throughput.

V. Performance Evaluation of the Proposed Solution

In this section, we design and develop a simulation model, which will allow us evaluate performance of our proposed routing system. We will use the simulation model to run several experiments under different conditions for both bb_path and ad_path.

A. Simulation Model

We implemented MANET in a mesh environment including backbone and access networks simulation model in OPNET modeler 11.5 PL1 [28] by creating a mesh of wireless routers in the backbone and clusters of MNs attached to each wireless router. We used 802.11a radio for backbone, and 802.11b/g for access network. 802.11a is used in the backbone because we believe the backbone is used to carry the major part of the traffic between clusters of MANETs, and it could become the bottleneck. Therefore it requires more bandwidth than the access network, which could be satisfied by 802.111b/g capacity. A campus network is deployed over a square geographical area of range 10*10 km2 as shown in Figure 4.

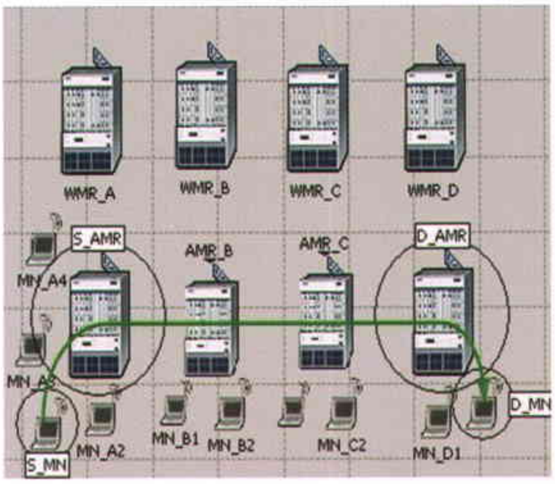

Figure 4: MANET in mesh environment implemented in OPNET, *bb_path* selected by S_MN to D_MN

MANET is deployed using eight WMRs creating the backbone, comprising of two rows of routers. The lower set of routers includes AMRs which connect MNs to the backbone. The upper set of routers is core WMRs that participate in the backbone, however do not have any MNs connecting to them for direct access purposes. First AMR in the lower row is named S_AMR, which depicts AMR corresponding to the source cluster, and the last AMR is D_AMR which shows the AMR corresponding to the destination cluster.

Each AMR is surrounded by MNs comprising a cluster for the corresponding AMR. For each cluster, we start the simulation with one MN, and then increase number of MNs to start the effect of increased traffic and channel contention.

Each AMR is equipped with two interfaces; one for the backbone running 802.11a and the other for access network running 802.11b/g according to Figure 4. At the initial stage, MNs have a single interface running 802.11b/g to connect to both AMR and ad-hoc network. The assumption is that MNs use same interface and same radio frequency to connect to both AMR and other MNs. This assumption is justified considering that all nodes are in the ad-hoc mode and capable of connecting to more than one peer at the same time. AMRs have 2 interfaces, one used for backbone communication with peer AMRs or WMRs, the other used for access network communication with MNs in the cluster. Since the backbone is on 802.11a, backbone traffic would not interfere with MN-MN and MN-AMR traffic. At the second stage, we turn on the second interface of MNs to be used for ad-hoc communication between MN-MN.

MANET traffic is generated from a S_MN to a D_MN according to the specifications in Table 1. Traffic is first generated from contending MNs in the cluster to go to the S_AMR. Then after 100 second when the traffic is continuously generated and contention is stabilized, then S_MN starts sending traffic to S_AMR. At this point new traffic is affected by the contention from other MNs.

The S_MN sends MANET traffic at exponential inter-arrival time of 0.01 second, and the constant packet sizes are 8,192 bits for the destination MN and 16,384 bits for the AMR. We set the throughput threshold at a minimum value of 100 bits/sec in order for the second route discovery to be triggered. In the OPNET simulation environment, 100 bps with 0.01 second inter-arrival time, ensures that the backbone is saturated, and it triggers the ah_path to be activated. The simulation ran for 4 minutes each time, and repeated 10 times for each experiment, and by choosing the seed option of 20 in OPNET, it will be equivalent to 200 times in each case. All the results are averaged over the entire number of repeats.

Table 1: Traffic parameters generated from S_MN to D_MN

Traffic Parameter	Value
Start time	100 (0 sec for contending MNs)
Inter-arrival time	0.01 sec
Packet size	8192 or 16384 bits (depending on destination)
Destination	Destination MN (D_MN), (AMR for contending MNs)
Stop time	End of simulation

B. *Basic Routing through ah_path versus bb_path*

Figure 5 shows the throughput of the bb_path under three different channel contention situations. System throughput is measured at the destination node. We increase channel contention by increasing the number of MNs in the source cluster.

We observe in Figure 5 that the throughput at the destination MN decreases while the numbers of MNs in the source cluster increases. This is due to contention surge as the number of MN increases at source cluster, and consequently packet drop rate will increase. This is verified by measuring the number of retransmissions in the source clusters, which also increases with the decrease in throughput. It illustrates the situation when high contention in AMR cluster renders the an_path to be the bottleneck of the bb_path. These results could also be confirmed with the end-to-end delay between S_MN and D_MN for 2 versus 6 MNs in the source cluster as illustrated in Figure 6.

Figure 6 shows that the delay will rise dramatically as the number of MNs increases in the source cluster. This is clearly due to the contention level increase in the source cluster.

Figure 7 shows the scenario with 4 MNs, where we allowed traffic from backbone or ad-hoc paths individually, and measured throughput for each case. This figure shows clearly that ah_path could improve performance when the bb_path is constrained by contention.

The corresponding delay results in Figure 8 clearly show that the system could decrease the delay significantly if the MN chooses to take the alternative ah_path over the congested bb_path. Figure 8 shows that the delay is reduced substantially when the MN switches from bb_path to ah_path.

C. *Throughput-aware AODV Routing*

MN evaluates the goodness of bb_path and ah_path based on throughput measurements and using this evaluation, it makes a decision which path to take. The evaluation and decision making processes are implemented in Algorithms 1 and 2. This part of the simulation is based on the changes in the core of AODV source code in OPNET. New AODV_enabled nodes should be aware of the throughput values for each path. Each AMR measures its link throughput to the next hop or next AMR (this value is saved as OWN_THROUGHPUT).

Based on the current implementation, S_MN will broadcast the RREQ. S_AMR will receive this RREQ and uses regular AODV to forward the RREQ hop-by-hop to the D_MN. D_MN will reply by unicasting a RREP message back to D_AMR and includes its throughput in the RREP. This is a one-way downlink throughput of D_AMR to D_MN, and not the throughput for the reverse path. D_MN also sets ROUTE_TYPE to AH. D_AMR receives this RREP, compares RREP_THROUGHPUT (this is recorded as INTERMEDIATE_THROUGHPUT) with the OWN_THROUGHPUT, and updates the RREP_THROUGHPUT with the smaller value. Every WMR along the way takes this throughput and compares it with its own link throughput, and updates the RREP with the smaller value. Since the throughput provided by the backbone links are usually higher than any access network throughput, the original link throughput coming from D_MN which represents the throughput of sub_path3 is likely smaller than any backbone throughput and will be selected as path throughput thus-far of mesh_path. Therefore, this throughput will have to compete with the throughput of sub_path1 and the smaller value of these two will get elected as the throughput for the route. At the same time D_AMR will also change the ROUTE_TYPE to BB, and after this point it will stay as BB and will not change again.

On the other hand, if the RREP_THROUGHPUT is less than the threshold throughput and the second route discovery is initiated, a second RREQ will go through the ah_path to the next MN, and uses regular AODV to reach D_MN hop-by-hop. Thus, D_MN will have a second RREQ from ah_path. D_MN will send a second RREP through ah_path, and a procedure similar to the one in the bb_path will be repeated, except that ROUTE_TYPE will always stay at ROUTE_TYPE_AH for this path. The throughput added to RREP on the ad-hoc path is the link throughput between the D_MN and the next hop (neighboring MN). Each MN along the way will compare this throughput with its own link throughput to the next MN and update RREP accordingly.

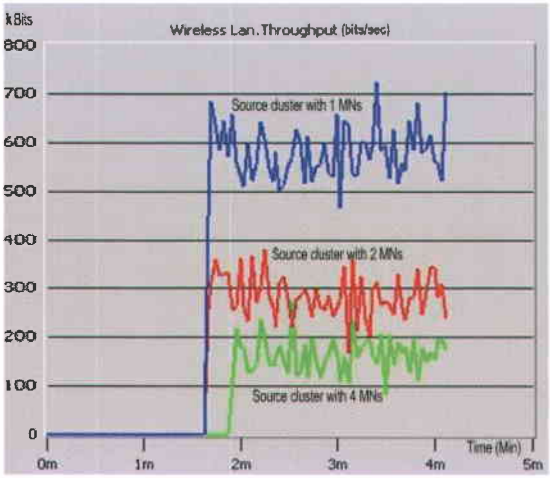

Figure 5: Throughput at the destination while increasing number of mobile nodes in a cluster resulting into increasing contention.

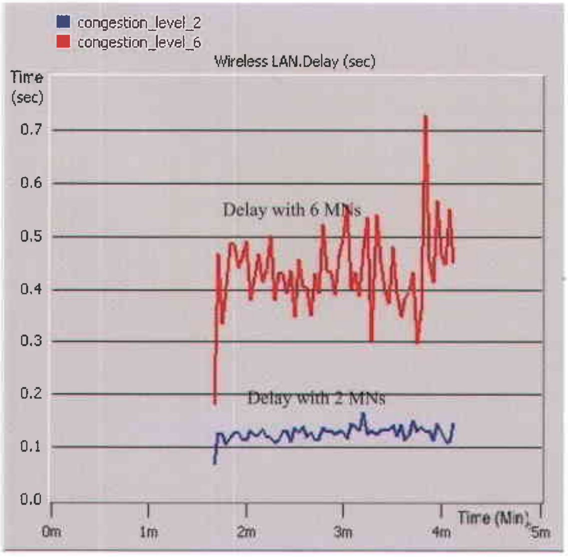

Figure 6: End-to-end delay for packets from S_MN to D_MN in presence of 2 versus 6 MNs in the source cluster

At this point S_MN will have two routes; BB and AH with each having its own throughput. S_MN will compare these two throughput values and uses the equation in algorithm 2 to decide which path to select. The AODV routing tables include two new columns for ROUTE_THROUGHPUT, and ROUTE_TYPE. The value of ROUTE_THROUGHPUT could be BB or AH throughput, depending on whether the node is an AMR or MN, respectively. The value of ROUTE_TYPE is a Boolean value (BB or AH, depending on whether the RREP is coming from an AMR or a MN). This is determined by extracting last digit of the IP address of the source in the RREP. All the AMRs are clusterheads, and their IP addresses are statically set to X.X.X.1, therefore, if the last digit of the IP address is 1, then the source is an AMR and the ROUTE_TYPE is set to BB, otherwise, it is set to AH.

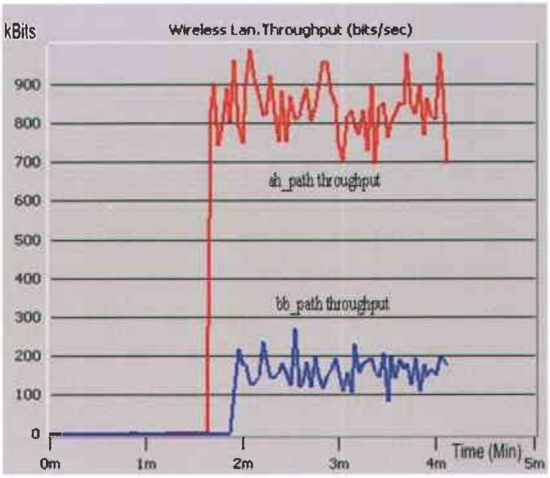

Figure 7: Throughput at the destination comparing *bb_path* versus *ah_path* for the case of high contention in the source cluster.

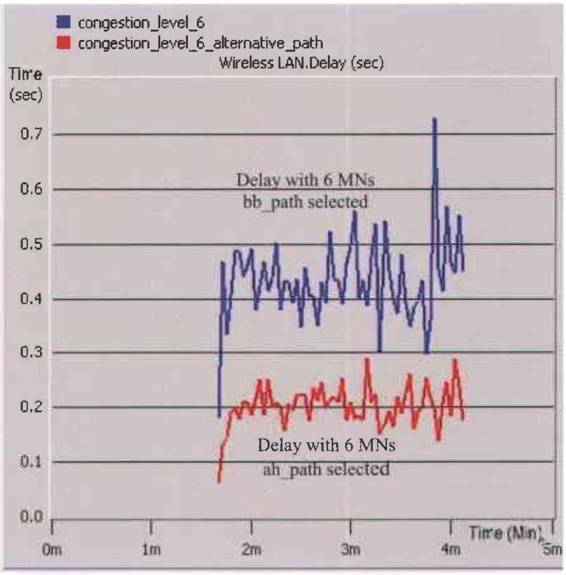

Figure 8: End-to-end delay for packets from S_MN to D_MN in presence of 6 MNs in the source cluster using ah_path versus bb_path

We changed AODV core by incorporating throughput in the routing cache and RREP packet of AODV, and implemented new source code in OPNET module. Then we compiled and ran simulation with the new source code. Once the MN receives the RREP packet, it is informed of the throughput values for the backbone, and it does a comparison with a threshold value for throughput. If the RREP reported throughput does not meet a minimum requirement set by the threshold, then the MN will switch to ah_path. In presences of 1 and 2 MNs in the source cluster, mesh_path is selected. By increasing number of MNs in the source cluster from 2 to 4, S_MN still chooses the mesh_path.

Figure 9 shows the scenario with 6 MNs in the source cluster. By increasing the number of MNs in the source cluster to 6, we observe from Figure 9 that the S_MN chooses the alternative ah_path rather than bb_path. S_MN favors the ah_path due to the fact that throughput performance is decreased below the minimum requirement set by algorithm 2.

Figure 10 shows the throughput results for the system when it switches from bb_path to ah_path by increasing the number of MNs from 2 to 6. Throughput in presence of 6 MNs has increased in comparison to selecting bb_path with 4 MNs. Initially, there is a drop in throughput. Then we observe a surge of over 4 fold. This indicates a switch from bb_path to ah_path. The increase is similar to that observed in Figure 7.

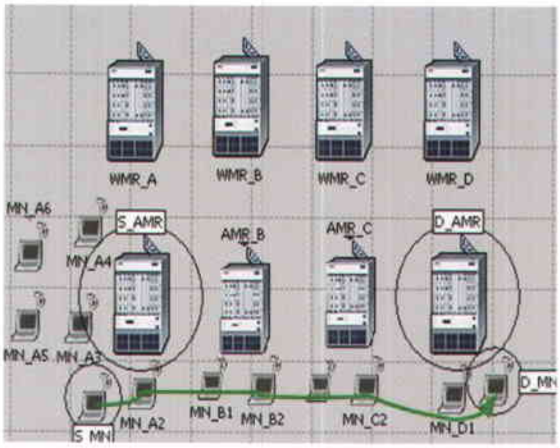

Figure 9: WMN with new AODV source code, 6 MNs in the source cluster and *ah_path* selected

The new improved performance surpasses that of both 2 and 4 MNs. However, eventually the new throughput comes back down to a level closer to 2 MNs, but still around 50% better than 4 MNs. This is due to the fact that initially next hop node for S_MN is still S_AMR, and S_MN still sends traffic via backbone. At this point there are still 6 MNs contending for the channel (contention level is 6).

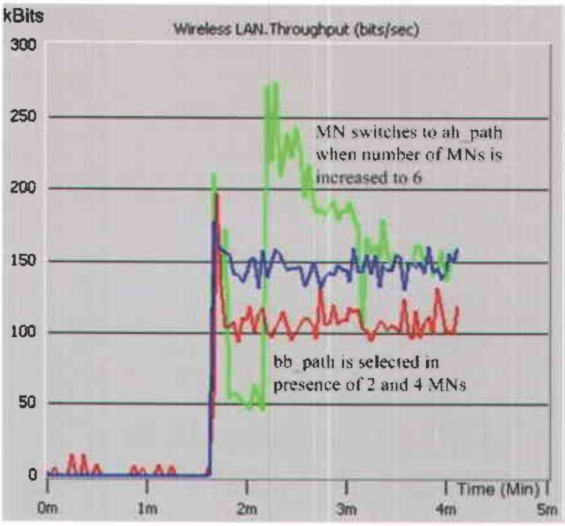

Figure 10: In presence of 6 MNs, throughput at the destination drops, MN switches path to ad_path, to compensate, return throughput to that of 2MN

After 130 seconds, we observe improvement in throughput. At this point traffic is switched and starts traversing via the ah_path and consequently the throughput will take a jump to above that of 2 MNs until around 200 second. This is due to the initial surge when switching to the ah_path takes place. After the initial switching surge, then throughput will drop to a stabilizing point which sits between throughput of 2 MNs and 4 MNs scenarios and continues at a steady rate beyond this point due to the fact that the number of MNs in the cluster remains constant, which was expected to experience similar contention levels; however the improvement in the throughput indicates that in case of ah_path fewer number of MNs are contending. In this scenario, it is observed that next hop for S_MN is MN_A2, and MN_A4 and MN_A6 are not visible from point of view of S_MN. This could be due to hidden-terminal issue, and indicates that contention level is 3, which means there are only 3 MNs contending for the channel. This is consistent with the throughput level in Figure 6 which sits between 2MN and 4MN scenarios.

Similar results could be observed from the delay performance measurements illustrated in Figure 11. As illustrated in Figure 11, delay increases significantly from 2 to 6 MNs in the source cluster while using bb_path. However, when the MN chooses the ah_path as an alternative, the delay drops significantly to almost that of 2 MNs.

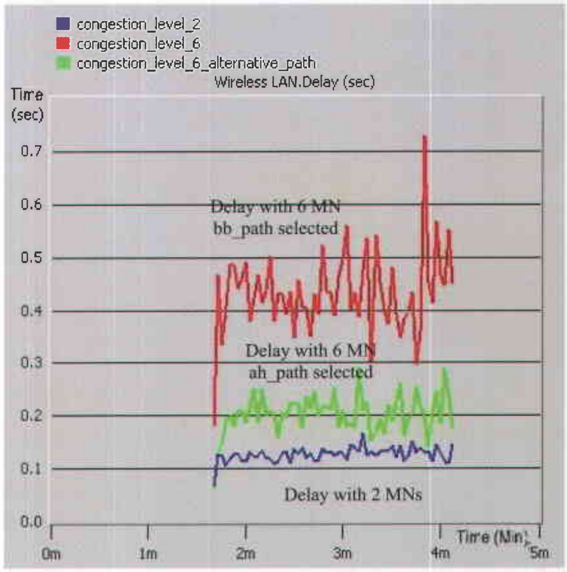

Figure 11: Delay measurements in presence of 2 MNs and 6 MNs using *bb_path* versus *ah_path*

VI. CONCLUSIONS AND FUTURE WORK

We proposed an integrated routing system for MANET in a mesh environment similar to WMN that exploits the paths through the backbone and the access networks. The ad-hoc path is considered as an alternative path and can be used under the following situations: (a) when the primary backbone path is severely constrained due to access links contention, (b) during handover to hide losses due to bb_path outage, and (c) for shortest path routing when bb_path is longer than ah_path. We have simulated the access contention situation and demonstrated the benefit of alternative ad-hoc path. We also proposed a scheme for initiating the route discovery of the ad-hoc path.

We incorporated throughput information in route cache of AODV and in a new field in the RREP packet of AODV, and allowed AODV to inform MN of the throughput information in addition to regular hop count. We also enabled MN to make a routing decision based on the throughput information.

In future, we plan to incorporate other link quality metrics (e.g. ETX) in AODV, and incorporate QoS metrics in the decision of using ah_path. We are also developing a routing based framework for mobility management in WMN that will use ah_path to hide the handover related losses and delay. We will develop our alternative ah_path solution using mobile nodes with two interfaces, and evaluate performance of WMN in presence of WMRs and MNs, both with two interfaces each.

REFERENCES

[1] F. Akyildiz, I. Wang, and X. Wang, Wireless Mesh Networks: A Survey, Computer Networks Journal (Elsevier), March 2005.

[2] D. S. J. De Couto, et al, A High-Throughput Path Metric for Multi-Hop Wireless Routing, ACM MobiCom Sept. 2003, pp. 134–146.

[3] R. Draves, J. Padhye, and B. Zill, Routing in Multi-radio, Multi-hop Wireless Mesh Networks, ACM MobiCom, Philadelphia, PA, 2004

[4] U.C. Kozat, I. Koutsopoulos, L. Tassiulas, A framework for cross-layer design of energy-efficient communication with QoS provisioning in multi-hop wireless networks, in: IEEE Annual Conference on Computer Communications (INFOCOM), 2004, pp. 1446–1456.

[5] R. Draves, J. Padhye, and B. Zill, Comparison of Routing Metrics for Static Multi-Hop Wireless Networks, SIGCOMM'04, Aug 2004,Portland, USA.

[6] L. Qin and T. Kunz, survey on MANET routing protocols and cross layer design, Technical Report SCE-04-14, Carleton University, 2006.

[7] L. Iannone, R. Khalili, K. Salamatian, and S. Fdida, "Cross-Layer Routing in Wireless Mesh Networks," Proc. of ISWCS 2004, September 2004.

[8] R. d. Renecsse, et al, QoS enabled Routing in Mobile Ad-hoc Networks. Technical Report. Electrical Engineering Department, Centre for Telecommunications Research, King's College London, UK.

[9] L. Iannone and S.Fdida, MeshDV: A Distance Vector mobility-tolerant routing protocol for Wireless Mesh Networks, IEEE REALMAN, 2005

[10] J. Wu, M. Jaseemuddin, and A. Esmailpour, Integrating UMTS and Ad Hoc Networks, Proceedings of IEEE Int. Conference on Wireless and Mobile Computing, Networking and Communications (WiMob 2006), 2006.

[11] D. Johnson, Y. Hu, and D. Maltz, "The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4," IETF,RFC 4728, Feb. 2007.

[12] D. De Couto et al, Performance of Multi-hop Wireless Networks; Shortest Path is not enough, the 1st Workshop on Hot topics in Networking (HotNets-I), 2002.

[13] E.M. Royer and C-K Toh, "A review of current routing protocols for MANET", IEEE Personal Communications, vol. 6 No 2, pp. 46-55. April 1999.

[14] S. Murthy and J.J. Garcia-Luna-Aceves, "An Efficient Routing Protocol for Wireless Networks", ACM Mobile Networks and App. J., Special Issue on Routing in Mobile Communication Networks, Oct. 1996, pp. 183-97.

[15] J. Broch, D. A. Maltz, D. B. Johnson, Yih-Chun Hu, and Jorjeta Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocols. In Proc. ACM/IEEE MobiCom, pages 85–97, October 1998.

[16] C. E. Perkins, P. Bhagwat. Highly Dynamic Destination-Sequenced Distance Vector (DTDV) for Mobile Computers Proc. of the SIGCOMM 1994 Conference on Communications Architectures, Protocols and Applications, Aug 1994, pp 234-244.

[17] P. JACQUET, P. MUHLETHALER, A. QAYYUM, A. LAOUITI, L. VIENNOT, T. CLAUSEN. OLSR: Optimized Link State Routing Protocol , RFC 3626, http://www.olsr.org/

[18] Ad-hoc On-demand Distance Vector - C. PERKINS, E.ROYER AND S. DAS Ad hoc On-demand Distance Vector (AODV) Routing, RFC 3561

[19] D. Johnson, D. Maltz, Y. Hu: The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks, Internet Draft, draft-ietf-manet-dsr-10.txt, work in progress, July 2004

[20] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, "Cross-layer design for wireless networks." WNCG Technical, Report TR-2003-04-00001, Wireless Networking and Communications Group, Department of Electrical and Computer Engineering, The University of Texas at Austin. Available for download at http://www.wncg.org.

[21] M. Chiang, "To layer" or "not to layer": balancing transport and physical layers in wireless multihop networks, in: IEEE Annual Conference on Computer Communications (INFOCOM), 2004, pp. 2525–2536.

[22] L. Iannone, K. Kabassanov, S. Fdida. Evaluation of Cross-Layer Rate-Aware Routing in a Wireless Mesh Network Test-Bed. EURASIP Journal on Wireless Communications and Networking Special Issue: Mobile Multi-Hop Ad Hoc Networks: From Theory to Reality, 2007(0), January 2007

[23] J. Moy, OSPF-Version 2, RFC 2328, April 1998

[24] Giuseppe Bianchi, Performance Analysis of the IEEE 802.11 Distributed Coordination Function". IEEE journal on selected areas in communications Vol. 18, No. 3, March 2000

[25] M. Kazantzidis, "Wireless Adaptive Multimedia using Network Measurements". UCLA CS Technical Report 200102, Feb 14, 2001

[26] Q. He et al., "Performance Comparison of Two Routing Protocols Based On WMN". Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007, page(s): 1726-1729

[27] S. Keshav, "A Control-Theoretic Approach to Flow Control", ACM Annual International Conference on Mobile Computing and Networking (MOBICOM), 1991

[28] OPNET technologies Network R&D solutions, OPNET modeler, available on Internet, 2008,

http://www.opnet.com/solutions/network_rd/modeler.html

[29] A. Esmailpour, M. Jaseemuddin, N. Nasser, and O. Bazan. "Ad-hoc Path: an Alternative to Backbone for Wireless Mesh Networks". Proceedings of IEEE International Conference on Communications (ICC), Glasgow, U.K., 24-28 June, 2007.

[30] Nortel network publication on-line. Available on the Internet in December 2008 at:

www.nortel.com/solutions/wrlsmesh/index.html

Amir Esmailpour is currently a Ph.D. candidate at the University of Guelph, Guelph, Canada. Amir received his Bachelor of Science degree at the University of Ottawa, and Master's of Applied Science from Ryerson University in Toronto, Canada. He worked at Nortel Networks as a Software Engineer and Daimler Chrysler as a Network Engineer for seven years, and returned to academic studies and research to pursue his PhD degree. His area of research is in Wireless Mesh Networks and Quality of Service (QoS) for the IEEE 802.16 standard. He is presently working on his Ph.D. thesis in Radio Resource Management and QoS for mobile WiMAX.

Dr. Nidal Nasser completed his Ph.D. in the School of Computing at Queen's University, Kingston, Ontario, Canada, in 2004. He is currently an Associate Professor in the Department of Computing and Information Science at University of Guelph, Canada. He is an associate editor of the Journal of Computer Systems, Networks, and Communications, Wiley's International Journal of Wireless Communications and Mobile Computing and Wiley's Security and Communication Networks Journal.