Design and Implementation of Two-tier Mobile Ad Hoc Networks with Seamless Roaming and Load-balancing Routing Capability\textsuperscript{1}

Yuan-Ying Hsu, Yu-Chee Tseng, Chien-Chao Tseng, Chi-Fu Huang, Jung-Hsuan Fan, and Hsiao-Lu Wu

Department of Computer Science and Information Engineering
National Chiao-Tung University
Hsin-Chu, 30050, Taiwan
yctseng@csie.nctu.edu.tw

Abstract

Due to its flexibility, Mobile Ad Hoc Network (MANET) has attracted a lot of attention recently. Most existing works, however, limit a MANET as a stand-alone network. In this paper, we propose a two-tier MANET by extending the connectivity of the MANET to the Internet. Some hosts in the MANET are equipped with cellular interfaces and are called gateways, which can provide Internet connections. Such extension would greatly improve the connectivity of MANET. To maintain seamless connectivity, we modify the Mobile IP to make traversing private networks (i.e., NAT) of cellular networks possible. We also propose a load-balancing routing protocol to utilize the cellular interfaces of gateways, which are likely to be bottlenecks to the Internet. A prototype is implemented and performance results obtained from the testbed is reported.

1. Introduction

With the advance of embedded computing technologies, portable devices, such as laptops, Personal Digital Assistants (PDAs), and cellular phones, have been widely used. A portable device usually has several wireless interfaces, such as IEEE 802.11 Wireless LAN (WLAN), General Packet Radio Service (GPRS), Personal Handy-phone System (PHS), and/or Bluetooth. Wireless communications are typically supported in two models: infrastructure and ad hoc, as illustrated in Fig. 1. Among these two options, forming a mobile ad hoc network (MANET) is more flexible since it is independent of the availability of base stations. Hence, intensive research has been dedicated to MANET [1, 2].

Figure 1: Two options of wireless networks: (a) infrastructure and (b) ad hoc.

A MANET is typically considered as a stand-alone network. However, it is important to enable its Internet accessibility. On one hand, users in a MANET can enjoy the tremendous resources in the Internet. On the other hand, the connectivity between multiple MANETs may be greatly improved. For such connectivity, several works [4, 5, 6, 7] have proposed possible architectures by deploying gateways to help mobile hosts route packets to the Internet. Among these approaches, some takes a proactive approach by modifying DSDV [4], some takes a reactive approach by modifying AODV [5, 6], while some takes a hybrid approach [7].

\textsuperscript{1}Y. C. Tseng's research is co-sponsored by the NSC Program for Promoting Academic Excellence of Universities under grant number 93-2752-E-007-001-PAE, by Computer and Communications Research Labs., ITRI, Taiwan, by Intel Inc., by the Institute for Information Industry and MOEA, R.O.C, under the Handheld Device Embedded System Software Technology Development Project and the Communications Software Technology Project, and by Chung-Shan Institute of Science and Technology under contract number BC93B12P.
C. C. Tseng’s research is co-sponsored by National Science Council under grant numbers NSC 92-2219-E-009-011 and NSC 92-2213-E-009-077, and by the Institute for Information Industry and MOEA, R.O.C under the Handheld Device Embedded System Software Technology Development Project.
In this paper, we propose a two-tier MANET architecture in which broadband WLANs (such as IEEE 802.11 a/b/g) are equipped in all mobile stations to form the low-tier network, and cellular interfaces (such as GPRS/PHS/3G) are equipped in some stations to form the high-tier network. Stations with high-tier interfaces are called gateways and can connect to the Internet. Such equipments are immediately available (e.g., the dual-interface card Nokia D211). Depending on its service range, each gateway together with the stations whose Internet connections are supported by the gateway constitutes a sub-MANET. The MANET under our scope is a collection of multiple sub-MANETs.

In this paper, we address several important design issues in such a two-tier MANET architecture. First, a dynamic IP address resolution is proposed. Second, observing that most cellular networks are considered private networks, causing Mobile IP [8] not usable, we propose how a station seamlessly roams between sub-MANETs without suffering from disconnection. In particular, we consider a MANET as a private network and propose NAT (network address translation) traversal mechanisms to support IP mobility under our two-tier architecture. Third, gateways in our environment are heterogeneous in nature (we allow mixture of GPRS/PHS/3G interfaces for gateways) and may become bottlenecks to the Internet. For such bandwidth-constrained interfaces, we propose a load-balancing routing mechanism to improve Internet access quality by allowing mobile hosts dynamically changing their gateways, thus relieving the bottleneck problem. Moreover, changing gateways should not break current connections.

We have successfully prototyped such a two-tier MANET architecture by properly modifying Dynamic Host Configuration Protocol (DHCP) and Mobile IP and integrating them under our load-balancing routing protocol. Section 2 introduces our network architecture, and Section 3 discusses several protocol design issues. Section 4 elaborates some implementation details. Some performance testing results are presented in Section 5, and conclusions are drawn in Section 6.

2. Network Architecture

We consider a set of mobile hosts called a MANETs. Each host is equipped with an IEEE 802.11 WLAN card, and these interfaces form the low-tier network. The Destination-Sequenced Distance Vector (DSDV) routing protocol [3], which adopts a proactive approach, is used on the low-tier network. A number of hosts are designated as gateways. Each gateway host is equipped with an extra cellular interface, such as PHS or GPRS, which enables the host to access infrastructure networks (and thus the Internet). Cellular interfaces with Internet access capability form the high-tier network. Note that these interfaces can be heterogeneous. The MANET can be physically connected (through the low- or high-tier network) or
disconnected. Each gateway together with the stations whose Internet connections are supported by it is called a sub-MANET. A set of stations that forms a connected component but does not have a gateway inside is disconnected from the MANET and is not allowed to connect to the Internet. Fig. 2 shows an example, where there are three sub-MANETs and one disconnected component.

Each MANET is considered as a private network, and each gateway is responsible for offering IP addresses to mobile hosts. We set up a DHCP and a NAT server in each gateway to assign and translate IP addresses. Inside MANET, routing is supported by DSDV. Via DSDV, mobile hosts also learn where their gateways are. When the network topology changes or when gateways change their points of attachment, handoff procedure may be taken. To support seamless roaming, we adopt Mobile IP with the co-located address mode [8]. Hosts rely on their home agents to maintain their connections while roaming. Note that the high-tier interfaces may also use private IP addresses, making traditional Mobile IP unusable. Several solutions [9, 10, 11] have been proposed to support IP mobility under private networks. We apply the NAT traversal mechanism [9] to achieve seamless roaming in private networks.

As mentioned earlier, we allow heterogeneous cellular interfaces in the high-tier network. The bandwidths of high-tier interfaces are much less than those of the low-tier interfaces. Therefore, we propose an intelligent gateway selection strategy to improve the transmission quality and achieve load balance. This is integrated into the DSDV routing protocol and the advertisement messages from gateways.

3. Protocol Design

Our design goal is to provide seamless connection to the Internet when mobile hosts roam between MANETs and to achieve load-balancing routing when mobile hosts have multiple gateways available. We address three issues: IP address resolution, mobility management under NAT structure, and load-balancing routing.

3.1 Address Resolution

A DHCP server is installed in each gateway. Before communicating with other hosts, a mobile host needs to retrieve an IP address from a gateway. To avoid confusion, we assign an exclusive section of IP addresses to each DHCP server. Note that we also allow a host to use its old IP address after roaming into a new sub-MANET. Taking Figure 3 for example, when a new mobile host n joins the MANET, it first broadcasts a DHCP_discover. Three nodes, j, k, and m, belonging to two sub-MANETs receive the request and help forwarding the request to their DHCP servers. To avoid the broadcast storm problem [12], the forwarding is done by unicast. This can be achieved by configuring each internal mobile host as a DHCP relay.

Figure 3: Address resolution and DHCP operations for host n to join the MANET. 
Fig. 4 illustrates the related DHCP message flows. The DHCP_Discover is forwarded by the DHCP relay to the DHCP server via DSDV routing. The DHCP server then replies a DHCP_Offer by including an available IP address. Note that the host may receive multiple offers from several servers. So the host will broadcast a DHCP_Request to notify all DHCP servers the IP address that it selects. After obtaining an IP address, the host will turn itself its temporary IP address with the same DHCP server. Called is using the same IP address. There is also a parameter, procedure will be executed to ensure that no other host replies a DHCP_Ack if the IP is still available. Afterward, the Duplicate Address Detection (DAD) procedure will be executed to ensure that no other host is using the same IP address. There is also a parameter, called lease, after which the mobile host has to renew its temporary IP address with the same DHCP server. After obtaining an IP address, the host will turn itself into a DHCP relay of the DHCP server that it selects.

**Figure 4: The message flow of DHCP in our architecture.**

### 3.2 Mobility Management in Private Networks

Mobile IP [8] can maintain seamless connection for mobile hosts while roaming. In Mobile IP, a mobile host can use a co-located care-of address (CCoA) or a foreign-agent care-of address (FACoA) if there is a foreign agent in the foreign network. If the latter approach is adopted, routing in the low-tier network would cause problem because all hosts will use the same FACoA. Besides, agent advertisement in a MANET may also cause the broadcast storm. Consequently, we adopt the former approach in this work, and use DHCP to provide CCoAs. When a mobile host retrieves a CCoA, it registers to its home agent. All packets destined to the mobile host hereafter will be tunneled to its CCoA by the home agent.

Recall that the MANET is configured as a private network in our architecture, and thus the Mobile IP tunnel from a home agent to a mobile host needs to cross two NAT interfaces, one on the low-tier and the other on the high-tier cellular interface. Unfortunately, the typical Mobile IP does not allow mobile hosts to roam into private networks because home agents can not tunnel packets to a private CCoA. The other reason is that the original IP-in-IP tunnels hide port information, making NAT servers unable to do the translation. To maintain correct tunnels to mobile hosts in private networks, we adopt a mechanism called NAT traversal [9]. It applies UDP tunneling, which encapsulates an extra UDP/IP header outside the IP payload, to provide port information to NAT servers.

Fig. 5 shows how NAT traversal works. There are four NAT servers and two gateways (one using GPRS and one using PHS). NAT1 is the NAT server in the GPRS core network, to which NAT2 is connected, and NAT4 is the server in the PHS network, to which NAT3 is connected. IP4_PV[0] denotes the IP address of a host A when it is in a public/private domain. For example, the GPRS gateway has two interfaces: IPNAT2_PV2 on the low-tier network and IPNAT2_PV1 on the high-tier GPRS network.

Consider a mobile host MN, which has a permanent address IP_MN_PB1 and a home agent (HA) with an address IP_HA_PB1, in a public network PB1. Suppose that MN moves into a MANET and obtains a CCoA IP_MN_PV assigned by the gateway NAT2. MN will send a registration_request to inform HA its new care-of address via its serving gateway (step 1). When forwarding the request, NAT1 and NAT2 will update their transition tables (steps 2 and 3). When HA receives this registration_request, it compares its source address against the CCoA that it claims. If they are not compatible, HA considers MN as inside a private network and registers IPNAT1_PB as MN’s care-of address (step 4). Later on, for each packet from CN destined to MN, an extra IP/UDP header will be added (step 5). Both NAT1 and NAT2 will translate the packet according to their DNAT transition rules (steps 6 and 7). Similarly, whenever MN wants to send a packet, it also encapsulates an IP/UDP header (step 8). The SNAT transition rules will then be used to forward the packet to HA (we omit the details here). On receipt of the packet, HA then decapsulates the IP/UDP header and forwards the packet to CN.

When MN roams into a different sub-MANET, we allow it to keep its care-of address. However, MN has to re-register with HA (this is not required in typical Mobile IP); otherwise, HA and the NAT servers will not have correct transition rules. For example, when MN moves to NAT3, a re-registration will be sent to HA (step 9).
3.3 Load-Balancing Routing

A load-balancing routing protocol is designed to help mobile hosts choose their gateways. We adopt the Minimum Load-Index (MLI) routing in [13] in our implementation. MLI tries to dynamically establish boundaries between gateways’ service ranges by taking the load indices of gateways and the traffic loads of hosts into account. This scheme is fully distributed and can be run by each host independently. Each gateway $g$ will periodically broadcast advertisement messages containing its current load index ($L_g$), which is the ratio of the traffic load of its high-tier interface to the maximum bandwidth of its high-tier interface. Each host $x$ should keep a record of the load index of its current serving gateway. When a host $x$ hears an advertisement from any gateway $g$, the following rules are executed:

1. If $x$ currently has no serving gateway, it chooses $g$ as its serving gateway by recording $g$’s current load index and setting the host leading to $g$ with the shortest distance as its default gateway. Then $x$ rebroadcasts the advertisement.

2. If $g$ is already $x$’s serving gateway, $x$ should update $g$’s index as necessary and rebroadcast the advertisement.

3. If $g$ is different from $x$’s current serving gateway, say $g'$, then $x$ will update $g$ as its serving gateway with a predefined gateway-switching probability $P_g$ only if $x$ has accepted $g'$ as its serving gateway for over a time period $r = \frac{T_g}{C_g} + \Delta_t \leq L_g \cdot \frac{T_g}{C_g}$, where $T_g$ is the traffic index of $x$, $C_g$ is the total capacity of the high-tier interface of $g$, and $\Delta_t$ is a predefined gateway-switching threshold.

The above steps are similar to a diffusion process. A gateway with less traffic load will extend its service range, while one with more load will shrink its service range. Note that in the above rule 3, we do not require $x$ to rebroadcast the advertisement, no matter $x$ accepts $g$ as its new gateway or not. We do this on purpose so as to achieve a slow diffusion. Otherwise, a gateway with a very low load will quickly take too many hosts and lead to a network fluctuating situation. As a result, the service ranges of gateways will be extended at most one hop in each advertisement. $P_g$ and $\Delta_t$ are designed with the similar purpose.

Taking Fig. 3 for example, there are three sub-MANETs with load indices 0.5, 0.8, and 0.6, respectively. Then $n$ will choose $C$ as its serving gateway according to rule 1. Suppose that $n$ has a very high traffic load and thus saturates sub-MANET $C$. Then this may force host $k$ to move to sub-MANET $B$.

4. Implementation Details

We have implemented a prototype in an environment with five IBM laptops, and each with an 802.11b NIC working in ad hoc mode to form the low-
tier network. Two of them are equipped two PHS handsets to form the high-tier network. The operating system is the Windows 2000 Advanced Server supporting the following services. First, we activate the “Internet Connection Sharing” service on gateways to share their Internet connections with other hosts. Second, the “DHCP Server” service is also opened in each gateway. Third, the “Routing and Remote Access” service is activated in each host to offer routing services and to drive the TCP/IP forwarding engine. Besides, once a non-gateway host obtains an IP address, it should enable the “DHCP relay agent” ability in this service to extend the DHCP service range in our MANET environment.

We also modify the Windows protocol structure to enable hosts to communicate with other hosts inside and outside the MANET. Figure 6 illustrates how we intercept and encapsulate packets from the TCP/IP layer destined outside the MANET. The encapsulated packets will be sent back to the TCP/IP layer and wait to be transmitted. Note that packets destined inside the MANET will not be intercepted.

Next, we introduce some tricks to manage routing tables. Recall that each mobile host has a home address and a CCoA. Since Windows picks the address with the same prefix as the default gateway as the source address for any IP packet without a route entry, we set each mobile host’s default gateway as the home agent of the host. As a consequence of this, packets destined outside the MANET cannot be routed correctly because the default gateway is not correctly set to the serving gateway of each sub-MANET. To remedy this problem, we purposely add a route entry in each mobile host by indicating how to route a packet to its home agent (by pointing to the host closer to the serving gateway than the host itself). This will enforce Windows route all IP/UDP-encapsulated packets as we expect.

As the last comment, we embed our MLI routing scheme in DSDV. In DSDV, hosts will exchange routing tables with neighbors. We create a MLI routing table with two more fields: gateway indication and loading. The former is to notify whether this routing entry is generated by a gateway or not, and the latter is to record the load index and capacity of the gateway. We wrote a MLI daemon for hosts to periodically exchange and update their MLI routing tables. Besides, the MLI daemon is also responsible for updating the system routing table that is actually used to forward packets in each host. Whenever the MLI daemon decides to change gateways, it updates the routing entry of its home agent in the system routing table.

5. Performance Evaluation

In this section, we present some performance results observed from our implementation testbed. The testing environment is shown in Fig. 7. In the beginning, there are two separate MANETs: N3 connected to gateway 1 (N1) and N4 connected to gateway 2 (N2). Then a new host N5 arrives, which can connect to both N3 and N4 and needs to communicate with a corresponding node (CN) in the Internet. We run a FTP session between N5 and CN. We also generate some random background traffic ranging from 2 Kbps to 8 Kbps from N3 and N4 to CN, and study the throughput between N5 and CN.

Fig. 8 (a) illustrates how N5 chooses its serving gateway, given the traffic loads of gateways in Fig. 8 (b). We observe that in general N5 can choose the lightest-load gateway without problem. However, there may be some delay before the right gateway can be selected because of the broadcast periods of gateways (here we set the period to be 10 seconds).

As the last comment, we embed our MLI routing scheme in DSDV. In DSDV, hosts will exchange routing tables with neighbors. We create a MLI routing table with two more fields: gateway indication and loading. The former is to notify whether this routing entry is generated by a gateway or not, and the latter is to record the load index and capacity of the gateway. We wrote a MLI daemon for hosts to periodically exchange and update their MLI routing tables. Besides, the MLI daemon is also responsible for updating the system routing table that is actually used to forward packets in each host. Whenever the MLI daemon decides to change gateways, it updates the routing entry of its home agent in the system routing table.

5. Performance Evaluation

In this section, we present some performance results observed from our implementation testbed. The testing environment is shown in Fig. 7. In the beginning, there are two separate MANETs: N3 connected to gateway 1 (N1) and N4 connected to gateway 2 (N2). Then a new host N5 arrives, which can connect to both N3 and N4 and needs to communicate with a corresponding node (CN) in the Internet. We run a FTP session between N5 and CN. We also generate some random background traffic ranging from 2 Kbps to 8 Kbps from N3 and N4 to CN, and study the throughput between N5 and CN.

Fig. 8 (a) illustrates how N5 chooses its serving gateway, given the traffic loads of gateways in Fig. 8 (b). We observe that in general N5 can choose the lightest-load gateway without problem. However, there may be some delay before the right gateway can be selected because of the broadcast periods of gateways (here we set the period to be 10 seconds).
Figure 9 traces the throughput of N5 when our load-balancing routing protocol is applied or not. The traffic being generated is the same as Fig. 8 (b). We see not only seamless handoff of N5 between gateways, but also quite stable throughput connecting to CN. On the contrary, the throughput of N5 fluctuates if N5 sticks to N1 or N2.

![Figure 9: Trace of N5's throughput over time.](image)

We mentioned earlier that the broadcast period of gateways may delay a host’s gateway selection. Here we set the broadcast period to 10 seconds and \( \tau = 0 \) second, but vary the traffic loads of N3 and N4 every 60, 90, 120, and 150 seconds. As shown in Table 1, when the background traffics fluctuate too fast (variation period below 90 seconds), N5 may not choose the correct gateway in time. The degradation of average throughput is due to handoff overheads (such as Mobile IP re-registration, during which no packet will be sent to CN, thus causing waste of bandwidth).

Table 1: The impact of traffic fluctuation to our load-balancing protocol.

<table>
<thead>
<tr>
<th>Variation period</th>
<th>60 sec.</th>
<th>90 sec.</th>
<th>120 sec.</th>
<th>150 sec.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Load-balancing</td>
<td>1176.42</td>
<td>1342.645</td>
<td>1446.227</td>
<td>1419.692</td>
</tr>
<tr>
<td>Stick-to-N1</td>
<td>1357.91</td>
<td>1445.480</td>
<td>1121.951</td>
<td>1377.744</td>
</tr>
<tr>
<td>Stick-to-N2</td>
<td>1400.30</td>
<td>1403.272</td>
<td>1286.676</td>
<td>1246.590</td>
</tr>
</tbody>
</table>

6. Conclusions

In the literature, most works consider a MANET as a stand-alone network. In this paper, we design a two-tier MANET, considering gateways as bridges to the Internet. This greatly improves the connectivity of MANET because cellular networks nowadays are almost globally available. However, Internet connectivity does cause several problems: address configuration, connection maintenance when roaming, and traffic bottleneck on gateways. In this paper, we show how to configure a MANET as a private network and modify DHCP to avoid possible broadcast storms. Mobile IP and NAT traversal are adopted to provide seamless roaming capability in private networks. A load-balancing routing protocol was implemented to relieve the bottleneck problem. A prototype is implemented on Windows platform to verify our architecture and testing results do show the benefit of load-balancing routing.

7. References