Inter-Domain Routing for Heterogeneous Mobile Ad Hoc Networks

***Abstract*—Inter-domain routing is an important component to allow interoperation among heterogeneous network domains operated by different organizations. Although inter-domain routing has been well supported in the Internet, there has been relatively little support to the Mobile Ad Hoc Networks (MANETs) space. In MANETs, the inter-domain routing problem is challenged by: (1) dynamic network topology due to mobility, and (2) diverse intra-domain ad hoc routing protocols. In this paper, we discuss how to enable inter-domain routing among MANETs, and to handle the dynamic nature of MANETs. We first present the design challenges for inter-domain routing in MANETs, and then propose a framework for inter-domain routing in MANETs.**

I. Introduction

Mobile ad hoc networks (MANETs) can enable effective communications in dynamic operation environments including a coalition military operation, emergency operation for disaster recovery, and on-the-fly team formation for a common mission, such as search and rescue. In these situations, multiple groups and organizations need to come together, communicate, and collaborate to achieve a common goal. For example, in a disaster recovery scenario, the local police force may need to coordinate with fire fighters, military forces, and medical crews by sharing information and communicating with each other regardless of the particular networking technologies that each group uses.

Another practical usage of MANETs in the near future is in the context of vehicular area networks (VANETs). In this scenario, groups of cars on the road will instantly form a communication network for sharing traffic information, preventing accidents, and data sharing. However, it is unlikely all cars will support the same network technologies, not to mention belong to the same network. The VANET for a particular car will be based on various factors such as auto manufacturer (who may employ a common network service for its own cars), service plans (people may subscribe to a network service plan of their own choosing), and other personal/business imperatives (employees of a company may be on the same network service). However, a single VANET may not be connected all the time and may only reach others via other VANETs. Such application scenarios call for development of a technology to enable end-to-end communications over heterogeneous MANETs governed by distinct administrative domains.

Facilitating interoperation among multiple MANETs presents a significant challenge at multiple levels, from physical to application layers. In this paper, we focus our investigation on the problem of inter-domain routing in MANETs. In the Internet, the Border Gateway Protocol (BGP) [6] provides a well-established mechanism for inter-domain routing among heterogeneous domains, called autonomous systems (AS). The principle of BGP is to enable *opaque* interoperation, where each domain has the administrative control over its intra-domain routing protocol and inter-domain routing policy, which is not known (or opaque) to the other domains.

Unlike in the Internet, the inter-domain routing problem is fundamentally different in MANETs with significant challenges. First, in MANETs, the network connectivity changes dynamically, thus an inter-domain routing protocol must be able to cope with such changes as network partitions/merges and connectivity changes. In addition, there are no clear boundaries between network domains and in many cases multiple domains may overlap in the same geographic region. Second, MANET environment has spawned out a new breed of routing protocols such as reactive routing protocols, geo-routing protocols, etc. [1] that are specialized for dynamic networks, and they require special handling to participate in inter-domain routing.

In this paper, we propose a novel networking framework, called IDRM (**I**nter-**D**omain **R**outing for **M**ANETs) to enable inter-domain routing between MANETs (and between MANETs and the Internet). IDRM has been designed to effectively address the two main challenges identified above. Particularly, it employs a proactive routing for inter-domain gateway communication to readily detect any topology changes (within a domain and among domains), and adapt to those changes. It supports each domain to participate in the inter-domain routing operation without any changes to their native intra-domain routing protocols. It also supports a policy-based routing in the same spirit as in the Internet to allow business relations and administrative control could be specified. This will allow a seamless integration of IDRM with the BGP when MANETs need to interoperate with the wired network.

II. Insufficiency of Extant Routing Frameworks

In this section, we discuss why extant routing frameworks are insufficient to support inter-domain routing in MANETs. Particularly, we explain why a BGP-like protocol is inapplicable in the ad hoc environment, and what in extant ad hoc routing frameworks are missing to support interoperation among multiple MANET domains.

A. Inadequacy of BGP for MANETs

Consider Figure 1, which consists of three MANET domains. One might apply a BGP-like protocol to this scenario as in Figure 2. However, there are several issues that render such a protocol inapplicable. First, the path vector protocol in BGP implicitly assumes the availability of the following functions:

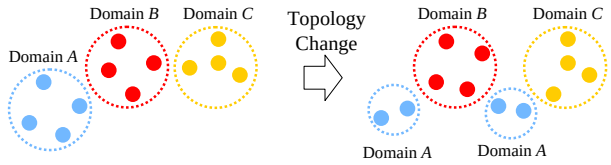

Fig. 1. The MANET of domain A is partitioned due to mobility.

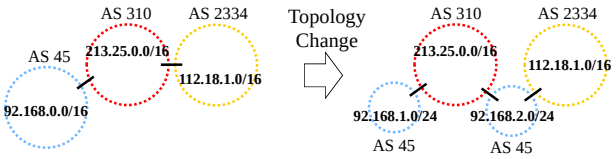

Fig. 2. A similar setting in terms of topology change in BGP.

(1) **Internal Gateway Detection:** The internal gateways within the same domain can detect the presence of each other so that they can communicate about the information of external routes.
(2) **Internal Network Knowledge:** The gateways know the reachable destinations and the internal routes to the destinations within the domain.

These functions are normally supported by the proactive intra-domain routing protocols through continual maintenance of network state information. However, we cannot always assume the availability of this information in MANETs that use a reactive routing protocol in their domains. Also a direct application of a path vector protocol over MANETs to support these functions may be undesirable to MANETs with dynamic node mobility and scarce wireless communication bandwidth.

Second, in BGP every destination is identified by an IP address, which follows a certain network hierarchy. To announce the destinations in a domain, gateways will aggregate the IP addresses in the domain by suitable IP prefixes (e.g., 92.168.0.0/16). However, in MANETs, mobility can create arbitrary network partition, unlike the perfect split of IP addresses as in Figure 2. Hence, IP prefixes do not suitably aggregate the IP addresses in partitioned MANETs and thus we cannot use the prefix-based routing of BGP.

Third, BGP relies on a path vector protocol that filters the paths consisting of repeated AS numbers to prevent looping. For example, in Figure 2, after topology change, the inter-domain level path from a source in AS 45 (92.168.1.0/24) to AS 2334 (112.18.0.0/16) is AS 45→AS 310→AS 45→AS 2334. This path will be filtered by the BGP path vector protocol, and hence it will prevent the nodes in AS 45 (92.168.1.0/24) from reaching AS 2334 (112.18.0.0/16).

In general, the design considerations for inter-domain routing in MANETs are fundamentally different from that of BGP. The main challenge of BGP is to cope with the extreme scale of the Internet; however, the scale of the network is not the main concern in MANETs since they will be relatively small due to physical/wireless, technical, and geographical constraints. Rather the main challenge here is to handle the constant changes in the network connectivity at both individual node level and at network domain level due to node mobility.

B. Insufficiency of current ad hoc protocols

In the literature, there are several proposals to enable interoperations among multiple wireless domains [4] [7]. Most of them only focus on high level architectures and provide a sketch of required components (e.g., translation of different naming spaces, and different protocols). While these related works have considered various issues regarding interoperation of multiple networks, none of them provided a specific solution for inter-domain routing between MANETs.

In the wireless context, there have been proposals to take advantage of heterogeneous routing protocols to adapt to network dynamics and traffic characteristics. For example, hybrid routing protocols (e.g. SHARP [5]) uses both proactive and reactive routing protocols to adapt the routing behavior according to traffic patterns. The basic idea is to create proactive routing zones around nodes where there are lots of data traffic, and use reactive routing in other areas. Since the main goal of hybrid routing is to improve the routing performance in a *single* domain via adaptation, it cannot support the interaction of multiple domains with different routing protocols.

Another related approach is cluster-based networking in MANETs [2]. The idea of cluster-based networking is to form self-organizing clusters and a routing backbone among cluster heads. In this way, cluster-based networks can use hierarchical routing and achieve a scalable routing solution in a single domain. Although cluster-based routing has a structural similarity to inter-domain routing, they are essentially addressing two fundamentally different problems. The goal of inter-domain routing is to support multiple domains with *autonomous* control; on the other hand, a cluster-based routing is applicable in a single domain with a full control over its clusters (e.g., on cluster formation and cluster head election).

III. Design of IDRM

In this section, we present the design of a networking framework called IDRM (Inter-Domain Routing for MANETs) to support opaque interoperations among multiple domains of MANETs. In this framework, each domain retains administrative control within its own domain while participating in collaboration. To enable inter-domain communications, IDRM requires special nodes as *gateways*. The role of gateways is more than just handling inter-domain routing; they need to bridge any technical seam that may exist between MANETs at physical, MAC, and network layers. However, the main focus of this paper is limited to the inter-domain routing functions of the gateways. A non-gateway node does not participate in the communication with the nodes in another domain. Thus multiple MANET domains may operate in the same region.

A. Design Issues

Now we explain the key design points of the IDRM. There are several issues that we need to handle: (1) partition and merge of domains, (2) membership announcement, (3) support for policy-based routing, and (4) data plane operations. The first two points are due to node mobility and dynamic topology, and the latter two are general issues with inter-domain routing with autonomy of each domain.

1) Handling Domain-level Topology Changes: As discussed in the previous section, one of the key challenges for inter-domain routing in MANET is dynamic changes of the network topology. In particular, a single domain may be partitioned into multiple MANETs due to node mobility and the gateways in the domain must detect the event. In a domain where the intra-domain routing protocol is proactive, this event will be eventually detected via route updates. For a domain with a reactive intra-domain routing protocol, however, this event may not be detected for a long time. To handle this problem, in IDRM, the gateways maintain soft state by periodically sending beacons to each other. The period of beacon can be adaptively set based on the mobility of the nodes and the rate of topology change.

After detecting a partition, the gateways in the same partition should generate a new MANET ID so that the new partition can be uniquely identified. By dynamically assigning a new ID, we can prevent the path vector routing algorithm from mistakenly considering the route via partitioned networks as a loop.[1] This computation should be performed independently at each gateway in the way that (1) all the gateways in the same partition to generate the same ID, and (2) the collision of IDs of different networks to be as low as possible. One way to achieve these goals is to use a pseudo random number generator to create a new ID using the IDs of all the gateways in the network as input. The gateways in the same partition use a simple hash function (e.g., MD5) to generate a random number, then prefix it by the domain ID to get a new MANET ID. We encode the domain ID in the new MANET to support a dynamic policy translation (as discussed in III-A3 and [**?**]). Conversely, when multiple partitioned MANETs come close and get re-connected, this condition should be detected by the gateways and a new ID for the merged MANET should be generated. This follows the same process as the case of network partitioning.

2) Membership Management and Announcement: Periodically gateways should advertise the IDs of the nodes that they can reach; for this the gateways need to collect the IDs of all the nodes in the MANET for advertisement of the membership to other domains. As we pointed out earlier, in MANETs we cannot rely on IP prefix for routing between domains due to arbitrary partitions and merges. There are two possible approaches to deal with the situation. First, the gateways can coordinate and reassign the node IDs so that each MANETs can have a unique prefix every time a topology change occurs. However, this will incur significant management overhead (e.g., to generate unique prefix, generate unique node IDs, to update name-to-ID mapping) and thus will only be useful when the new topology will remain unchanged for a relatively long time.

Second, a more practical approach to handle topology changes is to let the gateways in partitioned networks advertise the membership information, and this membership digest is used for inter-domain routing. For a reasonable size MANET with less than 1000 nodes, we find that a plain membership digest containing a set of node IDs (e.g., IP addresses) without any compression is better than a more scalable solution [**?**]. Obviously, the second approach (based on membership digest) can cope with network dynamics better and is more graceful when partitioned MANETs merge (by just merging the memberships). Hence, we employ the second approach in IDRM.

Keeping track of the non-gateway membership in a domain poses a similar challenge to network partition detection; in a reactive routing domain, a gateway may have a stale view of its membership, and can only discover the membership change when it has data to transfer. Although we can periodically perform a membership query, this can be potentially expensive. Thus instead, we let a reactive domain only initiate a membership query when there is an indication that its membership may have changed, e.g., a node in the membership digest cannot be reached, *and* a timeout period has passed.

3) Policy Support: Inter-domain routing policy is enforced in a similar same way as in BGP. By exchanging route updates (announcements and withdrawal) in a path vector protocol, inter-domain routing policies will be translated as the decisions of filtering and selecting routes at gateways. Using a path vector protocol, if a gateway a_1 in domain A is willing to provide a transit service to a neighboring domain B for a destination with node ID c_1, then a_1 appends its MANET ID to the route announcement of the selected path to c_1 and announces it to a connected gateway b_1 in domain B. Upon receiving the announcement, b_1 will decide if this path is more preferable than the current using path to c_1 based on its routing policy. If a new path is selected, b_1 will record the source of announcement as a_1 and distributes the announcement to other internal gateways in the MANET.

There are a variety of ways to specify routing policy rules. For example, in a next-hop-based policy specification, gateways will select paths only based on the next-hop domain in the route announcement (based on commercial relations like customer, provider, or peer). In a path-based specification, a domain will specify a complete ordered preference of all the acyclic domain-level paths; paths with higher rank are more preferable. In a cost-based specification, a domain will assign a numerical cost to every other domain as a subjective evaluation of the performance. Gateways will select the paths with the minimum total cost of all the downstream domains. In our design, we do not restrict the way inter-domain policies could be specified, but we assume that a next hop specification is used in our description.

One important issue to address in MANETs is that these routing policies are defined by network operators as static rules. Now in a MANET environment, a single domain may partition into multiple networks (e.g., a domain A breaks down into A_1 and A_2). Thus it is necessary to have a mechanism to automatically translate the original policy when such topology change happens. In [?], we have reported preliminary results on how to translate the static policies when a domain partitions under the next-hop-based specification and the cost-based specification. We refer the reader to [?] for more discussion

[1]It is possible to extend this basic protocol to include a leader election process and let the leader of a domain coordinate intra-domain operations (e.g., hierarchical beaconing among gateways, or MANET ID generation). But we do not discuss such schemes here for simplicity.

on this topic. In general, designing a mechanism to handle dynamic policy translation for MANETs is an interesting topic requiring further research.

4) Data Plane Operations: When a node sends data packets to an external destination (in another domain or in another partitioned network), it forwards the packets to one of the reachable intra-domain gateways. In a reactive domain, the sending node will first initiate a route discovery, and a gateway node that has a route to the destination will respond. In a proactive domain, the sending node will have a list of intra-domain gateways, and select one of them based on its own preferences. In either case, the gateway will first see if it is directly connected to the domain that contains the destination. If it is then it just forwards the packet; otherwise, it will forward the packets to a gateway connected to the destination domain based on the inter-domain routing information.

For incoming packets, the gateway performs a protocol translation and invokes the intra-domain routing protocol. In a reactive domain, the gateway will initiate a route discovery process if it does not already have the route in the cache. In a proactive domain, the gateway can determine if the destination is reachable from the local routing table.

If for some reasons the destination cannot be reached (e.g., the node may have been disconnected from any domain) IDRM does not provide feedback for unreachable destination. Following the design principles of the Internet, the problem should be handled at a higher layer. Although we only discuss proactive and reactive routing protocols in this paper, it is not difficult to see that this framework can support other types of intra-domain routing protocols (e.g., geo-routing and hybrid routing). Thus we do not present these cases in this paper.

IV. Protocol Specification

In this section we describe the routing protocol of IDRM in pseudo codes. We divide the functionalities of IDRM into two sub-protocols, namely external IDRM (e-IDRM) and internal IDRM (i-IDRM). The main duty of e-IDRM is to coordinate communication between gateways in different MANETs and perform route maintenance, while i-IDRM is used to handle communication between gateways within a single MANET.

For a gateway i, let $D(i)$ denote the orginial Domain ID of i, and let $G(D(i))$ denote the set of the intra-domain gateways in domain $D(i)$. Also, $G^{\mathsf{intra}}(i)$ denote the set of the intra-domain gateways which i has connectivity, and $G^{\mathsf{inter}}(i)$ denote the set of the inter-domain gateways which are within i's single hop range. Let $M(i)$ denote the set of intra-domain members to that i has connectivity.

A. e-IDRM

Pseudo code 1 is executed periodically to keep 1) broadcasting e-IDRM beacons to neighbour inter-domain gateways, 2) removing expired route entries in IDRM route table, 3) reading underlying ad-hoc routing table and update $M(i)$ and IDRM route table accordingly, and 4) sending route update information to other gateways if there are any route changes. The e-IDRM beacons are sent by broadcast so that new joining gateways can learn the neighbouring information immediately.

Pseudo Code 1 Periodic inter-domain rountines

```
1: if (timer > e-IDRM Beacon interval) then
2:     broadcast e-IDRM beacon
3: end if
4:
5: if (timer > route maintenance interval) then
6:     for all routes in route table do
7:         remove expired route
8:     end for
9:     read native ad-hoc routing table and update M(i)
10:    update IDRM route table according to M(i)
11:
12:    if (route table changed) then
13:        broadcast e-IDRM route update
14:        invoke i-IDRM to send route update (in pseudo code 5)
15:    end if
16: end if
```

For route maintenance, when there are changes in the IDRM route table, the gateway will send a route update message to $G^{\mathsf{inter}}(i)$ and rely on i-IDRM to distribute the route update information to $G^{\mathsf{intra}}(i)$. A route update message can consist of the combination of following information: 1) new route is added, 2) existing route is updated, 3) existing route is deleted, and 4) new MANET ID is generated. Notice that IDRM depends on the underlying ad-hoc routing protocol to provide $M(i)$, in other words, gateway does not probe the membership by itself.

Pseudo Code 2 Main inter-domain rountine

```
1: if (recv e-IDRM Beacon) then
2:     update G^inter(i) according to MANET policy
3:     update IDRM route table according to MANET policy
4:
5:     if (new gateways are added to G^inter(i)) then
6:         exchange IDRM route table with the new gateways
7:     end if
8:
9:     if (route table changed) then
10:        broadcast e-IDRM route update
11:        invoke i-IDRM to send route update
12:    end if
13: end if
14:
15: if (recv e-IDRM route update packet) then
16:    update IDRM route table according to MANET policy
17:
18:    if (route table changed) then
19:        broadcast e-IDRM route update
20:        invoke i-IDRM to send route update
21:    end if
22: end if
23:
24: if (recv i-IDRM route update packet) then
25:    broadcast e-IDRM route update
26: end if
27:
28: if (recv data packet) then
29:    handleDataPacket(pkt)
30: end if
```

Pseudo code 2 is the main routine which consists of the core e-IDRM funtions. When a gateway recevies an e-IDRM beacon or a e-IDRM route update, it will update its IDRM

route table. In here, a gateway detects a new inter-domain gateway, they will first exchange their IDRM route tables. Similar to pseudo code 1, route update messages will be sent to $G^{\mathsf{inter}}(i)$ when there are route changes, and propagation of the route update information for intra-domain gateway will rely on i-IDRM protocol. When the gateway receives a data packets, it will pass it to Function HandleDataPacket() in pseudo code 3.

Pseudo Code 3 Function HandleDataPacket(pkt)

```
1: if (destination is listed in native ad-hoc route table) then
2:     forward data by using native ad-hoc routing
3: else if (destination is listed in IDRM route table) then
4:     tunnel traffic by using IDRM
5: else
6:     drop packet
7: end if
```

Pseudo code 3 handles data packets received by IDRM gateways, and will be used by both e-IDRM and i-IDRM protocols. When a gateway receive a data packet, it will first check the destination appears in the native ad-hoc routing table or not. If the destination appears in the native ad-hoc route table, the gateway will using the native ad-hoc routing to forward the data packet. If the destination address is not shown in the native ad-hoc route table but is shown in the IDRM route table, the gateway will tuunel the data packet to the next gateway according to the entry in the IDRM route table. Finally the gateway will drop the data packet if the destination address is not shown in either of route tables.

B. i-IDRM

Pseudo code 4 Periodic intra-domain rountines

```
1: if (timer > i-IDRM Beacon interval) then
2:     send i-IDRM beacons to G(D(i))
3: end if
```

Pseudo code 4 is a simple routine to send i-IDRM beacons periodically to gateways within a MANET. The main propose of i-IDRM beacon is to maintain $G^{\mathsf{intra}}(i)$ at different gateways, which will be used in pseudo code 5 to generate new MANET ID when partitioned or merged happen.

Pseudo code 5 is the main routine for i-IDRM. The main functions of this routine is to 1) maintain $G^{\mathsf{intra}}(i)$, and 2) exchange route update between $G^{\mathsf{intra}}(i)$. As we describe in section **??**, $G^{\mathsf{intra}}(i)$ is used to generate new MANET ID when a domain is partitioned or merged. Once a new MANET ID is generated, a route update will be sent to $G^{\mathsf{inter}}(i)$. When i-IDRM receives an i-IDRM route update from other nodes in $G^{\mathsf{intra}}(i)$, it will update the IDRM route table and notify e-IDRM to propogate any updates to inter-domain gateways. In i-IDRM, any updates at IDRM route table will not send to $G^{\mathsf{intra}}(i)$ because other gateways in $G^{\mathsf{intra}}(i)$ will also received the route update message from the update originator. On the other hand, when i-IDRM receives an e-IDRM route update, i-IDRM will help e-IDRM to distribute the route update to $G^{\mathsf{intra}}(i)$.

Pseudo code 5 Main intra-domain rountine

```
1: if (recv i-IDRM Beacon) then
2:     update G^intra(i)
3:     if (G^intra(i) changed) then
4:         generated new MANET ID
5:         invoke e-IDRM to send route update (in pseudo code 2)
6:     end if
7: end if
8:
9: if (recv i-IDRM route update packet) then
10:    update IDRM route table according to MANET policy
11:    if (route table changed) then
12:        invoke e-IDRM to send route update
13:    end if
14: end if
15:
16: if (recv e-IDRM route update from e-IDRM) then
17:    send i-IDRM route update to gateways in G^intra(i)
18: end if
19:
20: if (recv data packet) then
21:    handleDataPacket(pkt)
22: end if
```

V. Overhead Analysis

In this section, we study the feasibility of the proposed IDRM protocol by estimating the message overhead incurred by the protocol. Our analysis aims to convey a basic picture of the estimated overhead, without involving the detailed steps of the protocol. We assume that no control packets are dropped or retransmitted, and the inter-domain routing policies are simple, so that gateways will not switch forwarding paths except when the paths are disconnected. Our analysis follows a similar approach for analyzing proactive and reactive routing protocols in [3].

Symbol	Defintion
N	Total number of nodes in a domain
G	The number of gateways in a domain
r	Transmission radius
$\bar{\nu}$	Average speed of a node
$\bar{E}$	Average number of links in a domain

First, consider a single domain with the parameters as defined in Table V. Assume that the mobility process of nodes is stationary and be confined to a bounded area. For a pair of nodes, if one node moves out of the transmission radius of other, then the link between them breaks. So, the average lifespan of a link is $\Theta\big(r/\bar{\nu}\big)$, and the average number of link breakages per second due to mobility is $\Theta\big(\bar{E}\bar{\nu}/r\big)$.

Since the mobility process of nodes is stationary where there is no net links are created or broken over time, the average numbers of link creations per second due to mobility in the domain is also $\Theta\big(\bar{E}\bar{\nu}/r\big)$. Hence, the average number of link state changes (creations or breakages) per second is $\Theta\big(\bar{E}\bar{\nu}/r\big)$. The control overhead of intra-domain routing protocols is determined by the number of link state changes.

Next, we estimate the overhead for proactive intra-domain routing protocols, reactive intra-domain routing protocols, and inter-domain routing protocol, respectively.

(1) **Proactive Intra-domain Routing Protocols:** Each node periodically broadcasts hello packets to its neighbours. Based

on the received hello packets, each node announces a new link-state/distance-vector packet that will be propagated throughout the MANET. Let λ^{hel} be the number of hello packets broadcast by each node per second. The total number of hello packets per second is $\lambda^{\mathsf{hel}}N$.

Since the average number of link state changes per second is $\Theta\big(\bar{E}\bar{\nu}/r\big)$, the total number of link-state/distance-vector packets per second broadcast is $\mathrm{O}\big(\bar{E}^2\bar{\nu}/r\big)$. This is an upper bound because optimized broadcast-based protocols (e.g., OLSR) normally requires less than $\bar{E}$ transmissions for each link-state/distance-vector packet to propagate throughout the network. Thus, the estimated number of control packets per second is:

$$\lambda^{\mathsf{hel}}N + \mathrm{O}\big(\bar{E}^2\bar{\nu}/r\big) \tag{1}$$

This is also the control overhead per second (at domain level) by IDRM to detect network partition and merging.

(2) **Reactive Intra-domain Routing Protocols:** IDRM requires beaconing among gateways to detect network partition or merging. The number of gateway pairs that will beacon each other is upper bounded by $\mathrm{O}(G^2)$. Let λ^{bea} be the beaconing rate between a pair of gateways. Then total number of beacons per second by gateways is $\mathrm{O}(\lambda^{\mathsf{bea}}G^2)$.

Let $\bar{L}$ be the average number of hops between a pair of nodes in the MANET. The number of link state changes per second for a path between a pair of gateways is: $\Theta\big(\bar{L}\bar{\nu}/r\big)$. Since each link state change will incur maintenance overhead in reactive routing protocols, it is reasonable to assume that the number of control packets is proportional to the number of link state changes and the beaconing traffic. Hence, the estimated number of control packets per second required by IDRM to detect network partition and merging is:

$$\mathrm{O}\Big(\lambda^{\mathsf{bea}}G^2\bar{L}\bar{\nu}/r\Big) \tag{2}$$

(3) **Inter-domain Routing Protocol:** Suppose there are m^{pro} domains running proactive routing protocols and m^{rea} domains running reactive routing protocols. Also assume each domain has the same parameters as in Table V. Note that the path vector protocol in IDRM behaves like a proactive routing protocols, but with different parameters. Let λ^{inter} be the number of inter-domain hello packets broadcast by each gateway per second in the path vector protocol. The total number of hello packets generated in the multi-domain MANET per second is $(m^{\mathsf{pro}}+m^{\mathsf{rea}})\lambda^{\mathsf{inter}}\bar{G}$, where $\bar{G}$ denotes the average number of gateways in each domain.

If a pair of intra-domain gateways stay in the same MANET, there may be multiple paths connecting them. Let $1/\mu$ be the average lifespan of the connectivity between a pair of intra-domain gateways. That is, μ is the connectivity breakage rate of connected pairs of intra-domain gateways due to mobility. By stationarity of mobility process, μ is also the rate of change for the connectivity status of intra-domain gateways. Since IDRM will carry out new membership management and announcement when the connectivity status between a pair of intra-domain gateways is changed, the estimated number of connectivity status changes is:

$$\mathrm{O}\Big((m^{\mathsf{pro}}+m^{\mathsf{rea}})\mu\bar{G}^2\Big)$$

Hence, the total number of control packets per second for path vector protocol is:

$$(m^{\mathsf{pro}}+m^{\mathsf{rea}})\lambda^{\mathsf{inter}}\bar{G} + \mathrm{O}\Big((m^{\mathsf{pro}}+m^{\mathsf{rea}})\mu\bar{G}^2\bar{E}^{\mathsf{inter}}\Big) \tag{3}$$

where $\bar{E}^{\mathsf{inter}}$ is the average number of pairs of connected inter-domain gateways in the $(m^{\mathsf{pro}}+m^{\mathsf{rea}})$ domains.

In a given network, $m^{\mathsf{pro}}, m^{\mathsf{rea}}, \bar{G}, \bar{E}$, and λ^{inter} are fixed. It is not straightforward to decide μ. But we can obtain this value from simulation. In Figure 3, we observe μ decreases as the number of nodes increases because as a MANET becomes denser, the connectivity between a pair of gateways becomes more stable, whereas node speed adversely affects the stability of links almost linearly.

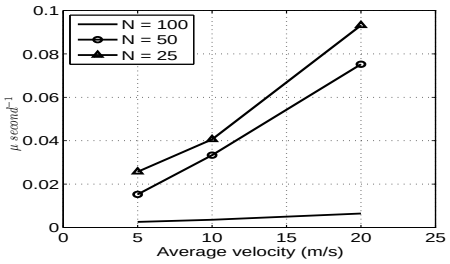

Fig. 3. Average lifespan of the connectivity between a pair of intra-domain gateways.

Note that Eq. (3) only provides an asymptotic result for the *total* control overhead incurred by IDRM without any optimization. Since the overhead will be distributed among all the gateways and various optimization can be applied (e.g., suppression of hello, adaptive adjustment of probing interval), the overhead incurred at each gateway for inter-domain routing operation will be quite moderate.

We compare this estimation to the normal routing overhead (not incurred by inter-domain routing). The overhead for proactive domains is Eq. (1), and the same for reactive domains is Eq. (2). Note that N and $\bar{E}$ are typically orders of magnitude greater than the other parameters. Thus, the overhead from reactive domains and inter-domain operations are substantially small compared to proactive domains.

To summarize, in a multi-domain MANET consisting of proactive and reactive domains, the overall control overhead is dominated by that of the proactive domains, and the overhead incurred by inter-domain routing protocol is relatively insignificant. Thus we report that inter-domain routing can be supported with moderate additional overheads in MANETs, and IDRM is a viable approach to enable that.

VI. Performance Evaluation

We evaluate the effectiveness of IDRM in three different scenarios. The first scenario is a stationary grid topology, and the second one is a 2-domain topology with simple group mobility. The propose of these two simple scenarios is to demostrate the basic characterisics of IDRM. The last scenario is random topology with random waypoint mobility, which can help us to understand the average performance of IDRM in complex scenario settings.

A. Simulation Setup

We modifiy ns-2 [**?**] in the way that it can support mutiple wireless interfaces in a wireless node. For each set of simulation, the communication range is 250 meters; we average our results from 5 simulations, each last for 2000 seconds simulation time.

We assume that everything nodes in a domain, say domain A, use $Channel_A$ to communcate. Non-gateway nodes have only one wireless interface. Gateway nodes have two wireless interface, one for domain specific ad hoc routing protocol and the other for IDRM. All IDRM traffic uses $Channel_{IDRM}$ to communcate. We also assume that channels are independent of each other, i.e., nodes in domain A cannot overhear traffic from domain B and non-gateway nodes cannot overhear IDRM traffic. We believe that our assumptions are practical because 1) different MANETs may use different technologies and coding, and 2) gateways are ususally stragetically decided before deployed so that they can afford multiple interfaces.

B. Scenario 1: Stationary grid topology

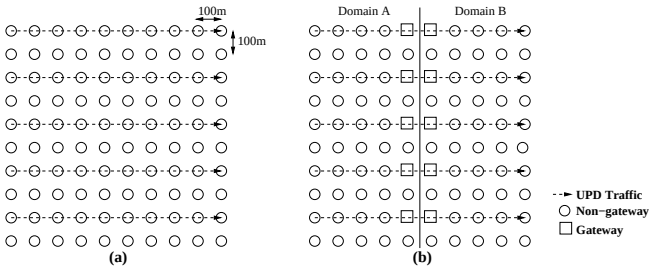

Fig. 4. Stationary grid topology for: (a) DSDV, and (b) IDRM

We first study the performance of IDRM with stationary grid topology shown in figure 4. The goal here is to demostrate the performance of IDRM with multiple domains is comparable to the performance of the single routing protocol with a signle domain. In figure 4a, all nodes are communicate through one channel, while in figure 4b, nodes are using 3 independent channels (one for domain A, one for IDRM, and one for domain B).

C. Scenario 2: 2-domain topology with simple group mobility

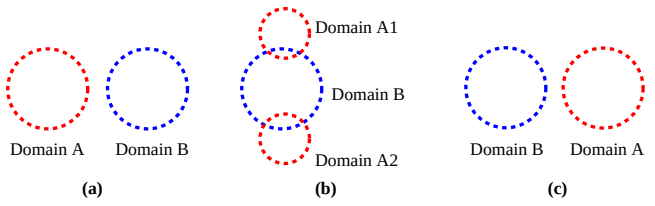

Fig. 5. Relative position of domain A and B at (a) Initial, (b) domain A is partitioned, and (c) domain $A1$ and $A2$ are merged

In this section, we study the performance of a simple group mobililty with 2-domain topology shown in figure 5. There are two domains in this scenario, namely domain A and domain B. We create a synthetic group mobility for domain A as follow: 1) Initially, domain A is located at the west side of domain B, 2) After a while, domain A is moving towards the east side of domain B and partitioned into two sub-domains $A1$ and $A2$, 3) Finally, $A1$ and $A2$ are merged back to domain A and domain A is located at the east side of domain B. Two UDP traffics, $U1$ and $U2$, are created for the evaluation. $U1$ is a traffic where the source is located in domain $A1$ and the destination is located at domain $A2$. During domain A is partitioned, $U1$ need to use the path $A1 \rightarrow B \rightarrow A2$ to maintain connectivity. For $U2$, the source is located at domain A and the destination is located at domain B.

D. Scenario 3: Random Topology

In the previous examples, we illustrate the effectiveness of IDRM with two simple scenarios. In this sections, we consider a more complex scenario with random topology. We randomly deployed 60 nodes in a 1000x1000 m^2 area, and the random waypoint mobility mode is used. Two average speed settings, 1.34m/s and 13.4m/s, are used to simulate walking and driving speed. Also, we study the performance of IDRM with different percentage of gateways and different number of domains.

E. Performance Metrics

For each scenario, we will study the following metrics. So the following will be merged into the above 3 scenarios

1) Network Reachability: We measure the network reachablity with different control parameters. We quantify the network reachability as the number of nodes that each node can reach at a moment. We assume the more number of reachable nodes means the better network reschability. We control the number of gateways and the interval of control packets.

- x-axis: time, or different parameters
- y-axis: Average number of reachable node for each node
- *Expected Results*: Each gateway is able to logically reach to all of nods physically reachable through other gateways.

2) Route Convergence: We measure the delay of route update messages over the entire network. The delay starts when a route update message is generated by a node and ends when every gateway reachable to the node receives the update message. We control the interval of control messages, communication range, and mobility for the evaluation.

- x-axis: Beacon interval time, different node speed and communication range
- y-asix: Average Route Convergence time

3) Route Consistence: We define a consistent route as a route entry that every gateway reachable to the entry has the same status of the entry. Because of network dynamics, it is possible that some of gateways have a valid value to the invalid entries. We control the interval of control messages, communication range, and mobility for the evaluation. As we have the global information in ns2, [Difficult to do] present the

4) UDP Throughput and Delay: We randomly generate disjoint source and destination pairs of UDP traffics and measure the throughput, and end-to-end delay of the traffics under network dynamics.

- x-axis: time, series: different parameters
- y-axis: UDP throughput, end-to-end delay

5) Overhead: We define the overhead of IDRM as the number of/ and the amount of IDRM control packets. We plot the percentage of each type of packets: data, IDRM packets.

- *Expected Results*: As tested with a small topology(two domains, 6 nodes), the overhead of IDRM was less than 2% and rest of packets were data.

VII. Discussion

A. Separation of inter and intra functionalites

B. Optimization for different ad hoc routing protocol

VIII. Conclusion

Inter-domain routing offers a means for heterogeneous MANETs to interoperate with each other. This paper has identified the challenges of inter-domain routing in MANETs, and proposed IDRM as a viable solution. This paper has shown that, despite dynamic network topology and diverse intra-domain ad hoc routing protocols, opaque interoperation among heterogeneous multi-domain MANETs can be supported. We also discussed how IDRM can support network operators to specify inter-domain routing policies in a similar manner as BGP. This is an important feature to encourage interoperation among multiple MANET domains in practice. We expect that IDRM will improve the end-to-end reachability of mobile users, and consequently enhance the usefulness of MANETs.

References

[1] M. Abolhasan, T. Wysocki, and E. Dutkiewicz. A review of routing protocols for mobile ad hoc networks. *Ad Hoc Networks*, 2:1–22, 2004.

[2] Y. Chen, A. Liestman, and J. Liu. Clustering algorithms for ad hoc wireless networks. In *Proc. Ad Hoc and Sensor Networks '04*, 2004.

[3] T. Clausen, P. Jacquet, and L. Viennot. Analyzing control traffic overhead versus mobility and data traffic activity in mobile ad-hoc network protocols. *ACM Wireless Networks journal (Winet)*, 10(4), July 2004.

[4] J. Crowcroft, S. Hand, R. Mortier, T. Roscoe, and A. Warfield. Plutarch: an argument for network pluralism. *ACM Computer Communication Review*, 33(4):258–266, 2003.

[5] V. Ramasubramanian, Z. J. Haas, and E. G. Sirer. SHARP: A hybrid adaptive routing protocol for mobile ad hoc networks. In *Proc. ACM MOBIHOC*, June 2003.

[6] Y. Rekhter and T. Li. RFC 1771: a Border Gateway Protocol 4 (BGP-4), March 1995.

[7] S. Schmid, L. Eggert, M. Brunner, and J. Quittek. TurfNet: An architecture for dynamically composable networks. In *Proc. of 1st IFIP International Workshop on Autonomic Communication (WAC 2004)*, October 2004.