Geographically Informed Inter-Domain Routing

Ricardo Oliveira* Mohit Lad* Beichuan Zhang† Lixia Zhang*

*{rveloso,mohit,lixia}@cs.ucla.edu †bzhang@cs.arizona.edu
University of California, Los Angeles University of Arizona

***Abstract*— In this paper we propose a new routing protocol and address scheme, Geographically Informed Inter-Domain Routing (GIRO). GIRO departs from previous geographic addressing proposals in that it uses geographic information to *assist*, not to replace, the provider-based IP address allocation and policy-based routing. We show that, by incorporating geographic information into the IP address structure, GIRO can significantly improve the scalability and performance of the global Internet routing system. Within the routing policy constraints, geographic information enables the selection of shortest available routing paths. We evaluate GIRO's performance through simulations using a Rocketfuel-measured Internet topology. Our results show that, compared to the current practice, GIRO can reduce the geographic distance for 70% of the existing BGP paths, and the reduction is more than 40% for about 20% of the paths. Furthermore, encoding geographic information into IP addresses also enables GIRO to apply geographical route aggregation, and a combination of geographic and topological aggregation can lead to 75% reduction of the current BGP routing table size.**

I. Introduction

The Internet consists of a large number of individually administrated networks called autonomous systems (ASes). The global routing protocol, BGP [18], is a path vector protocol that propagates routing reachability information among all the ASes. The route selection decisions are primarily driven by routing policies which reflect the Internet service providers (ISPs) economical interest. For example, when multiple routes exist to reach the same destination network, an ISP typically prefers a route going through its customers over those going through other providers. However, because the Internet topology is densely connected, even after applying the policy policies, a BGP router is often left with multiple feasible routes to reach a given destination network. Ideally, if everything else being equal, a router should choose a path with the shortest physical distance to optimize the data delivery performance. Unfortunately, in today's routing system, a router has information regarding the physical location of the destination or the distances of the alternative routes. In fact, several previous studies have shown ample evidence that the data paths used in today's Internet can be significantly longer than possible alternative paths as measured by their geographical distances [19], [23], [20].

In this paper we propose a new routing and addressing scheme called Geographically Informed Inter-domain Routing (GIRO) that aims to improve routing performance and scalability by adding geographical location information into the IP address structure. More specifically, GIRO aims to provide better global routing while still adhering to the policy based routing practice in today's Internet operations. In addition, including geographic location information in IP addresses opens the door to route aggregation based on geographic location. We show that such a route aggregation scheme can potentially lead to significant routing table size reduction.

To provide quantitative evaluation on the feasibility and effectiveness of the GIRO design, we first conducted measurement studies to map the existing prefixes in today's global routing table to their corresponding geographic locations. Using an Internet topology model extracted from Rocketfuel [21], we then converted the existing prefixes to the corresponding prefixes in a GIRO network. On this topology, we simulated the routing decisions following today's BGP policy practice with the enhancement of geographic location information and the physical distance information derived from it. Our evaluation results show that, compared to today's BGP path lengths, the GIRO design can reduce the routing path lengths by more than 40% for about 20% of all the paths. In addition, we show that embedding geographic location information in IP addresses enables a new *shortest-path* route selection, which can select shorter routing paths between neighbor ASes in about 30% of cases compared to today's early-exit BGP routing policy. Finally, we show that by applying a combined geographic and topological aggregation which is enabled by our new GIRO addressing structure, GIRO can shrink the BGP table to 25% of its current size.

The rest of the paper is organized as follows. Section II describes drawbacks in the current inter-domain routing system, presenting prior proposals in the area of incorporating geographical information in IP addresses, and highlighting the main differences between the past work and our proposed design. Section III presents the architecture of GIRO, including its address content and path selection process. Section IV describes our evaluation results. We discuss remaining open issues in GIRO design in section V, and related work in section VI. Finally, section VII concludes the paper.

II. Background and Motivation

A. Suboptimal Path Selection in Current Inter-domain Routing

BGP is a path vector protocol and routing information is propagated by the exchange of BGP update messages. A BGP update message contains information about the destination prefix and the AS path used to reach that prefix. Route selection and announcement in BGP are determined by networks' routing policies, in which the business relationship between two connected ASes plays a major role. AS relationship can be generally classified as customer-provider or peer-peer [1]. Usually a customer AS does not forward traffic between its providers, nor does a peer AS forward traffic between two other peers. When ASes choose their best path, the order of preferences is customer routes, peer routes, and then provider routes. This policy of *no-valley-prefer-customer* is generally followed by most networks in the Internet [9].

When given multiple routes with same policy preference, BGP breaks ties by picking the route with the lowest AS hop count. Figure 1 shows an example extracted from BGP log data. In this example, AS6461 is a peer of both AS3561 and AS577 and treats the two peer routes with equal preference. To reach AS577 in Seattle,WA from AS6461's location in Palo Alto,CA, AS6461 picked the route with a single AS hop 577 through Chicago following the minimal AS hop path selection guideline, since the alternative route through Seattle has two hops 3561-577. However, the route through Chicago spans a total distance of 3,584 miles, while the route through Seattle has 703 miles, a difference of more than 2,800 miles. A longer physical distance leads to higher latencies thus degrading end-to-end performance. One measurement study reported that about 75% of paths suffer inflation up to more than 15 msec [20], mainly caused by the use of AS hop count as a tie-break metric in the BGP decision process.

B. Desired Properties for an Inter-domain Routing Protocol

An inter-domain routing protocol must first be able to choose routes that satisfy given routing policies. The relationship between neighbor ASes determines which path is most preferred if multiple choices exist. Within the policy constraints, the routing protocol should choose the routes that offer good data delivery performance. The performance can be measured either within an ISP (e.g., by the link metric), or end-to-end (e.g., by end-to-end delivery delay). Both measures are important. ISPs desire good local performance that can minimize their cost in forwarding data traffic, as well as good end-to-end delivery performance to attract end users.

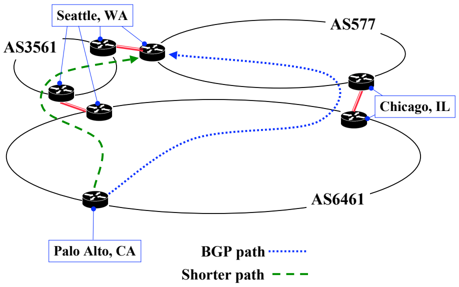


Fig. 1. AS6461 have a peer-to-peer relationship with AS577 and AS3561. The BGP route through Chicago travels a distance of 3,584 miles, whereas the shorter route through Seattle has a geographic length of 703 miles, shorter by more than 2,800 miles.

Due to the ever increasing density of AS interconnectivity, a router usually has multiple alternative paths to choose within its policy constraints. In the current practice, BGP chooses the path with minimum AS hop count first, which can be seen as an attempt to improve end-to-end performance. Among paths with the same AS hop count, BGP follows a multi-step decision process to nail down the final choice, and one important step is to choose the path with minimum IGP cost, which can be seen as an attempt to minimize AS internal cost.

However using AS hop count is very gross grained. Previous work has shown that the actual BGP paths can be significantly longer than the shortest policy-compliant path [19], [23], [20].

In this paper we propose to add geographic location information into IP addresses and use geographic distance information to help achieve the desired performance goals in route selection.

C. Previous Efforts in Geographic Addressing

The idea of incorporating the geographic location information into IP address structure is not new. It was first proposed by Finn in 1987 to address routing scalability issue, making addresses more aggregatable and enabling routing based on geographic distances[6]. A revised version, named "metro-based addressing" was proposed by Deering in early 90's as a solution to scalable multihoming and renumbering avoidance [5]. More recently yet another location-based addressing scheme, dubbed "Geo-based addressing" [10], was proposed. Although there exist certainly differences among these proposals, they all bear the fundamental notion of allocating IP addresses solely based on locations. There has been a fair amount of resistance to these proposals, because

[1] Sometimes the relationship between two AS nodes can be "siblings," usually because they belong to the same organization.

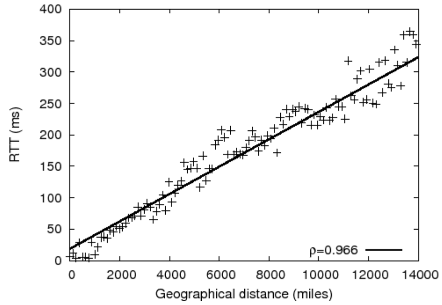


Fig. 2. Correlation between RTT and geographic distance using traceroute data from more then one hundred PlanetLab nodes.

location-based addresses do not reflect either the ownership of the addresses, nor the interconnectivity among network providers. As a result, routing based on geo-addresses not only requires that ISPs interconnect at all locations but also is unable to support routing policies. The limitations of geo-based addressing come from the lack of provider information in the address structure.

D. Incorporating Geographic Information into Addressing and Routing

An economically viable design must take provider economic interests into account as *first* priority to address the issue of “facilitating the routing of money”. Replacing the current address allocation with location-based approach, even partially, is not a feasible approach. However, an address structure that contains provider information as a first priority could be enhanced with location information to open the door to a wide variety of new routing functionality and policy support, which is the approach explored in this study.

Under the policy constraints, path selection can be better done with location information. Instead of minimum AS hop count, we can choose paths with shortest end-to-end distances. Due to the rich connectivity in the Internet topology, geographic distance has good correlation with end-to-end delay, as shown in Figure 2. Short end-to-end delays can provide several benefits to applications: (1) good performance for interactive, real-time applications, and (2) higher TCP throughput for non-realtime applications. Traversing longer distance (and more routing devices) can also increase the chance of outage, delay jitter, congestion and packet lost, hence it should be avoided.

Geographic location information can also help routers in choosing egress points within a domain, providing an alternative to hot-potato routing[24]. Our results also show that the “shortest path” policy selects the shortest end-to-end route without sacrificing much the intra-domain cost.

Furthermore, embedding both AS ID and geolocation information in a prefix opens the door to more effective route aggregation. E.g., prefixes originated from the same network might be aggregated according to geography.

III. GIRO Architecture

We now present our design and explain how we incorporate geographical information into the address structure and route selection process. Later in this section, we also explain how this geographical information can be used to achieve shorter routes, better egress point selection as well as smaller global routing tables.

A. Addressing in GIRO

In order to incorporate geographical information into the address structure we define a new address format called GIRO address. A GIRO address has two distinct components: *external* and *internal*. The external component consists of (1) its network ID in the form of **AS number (ASN)**, (2) its **geographical location (geolocation)**, and (3) its **traffic slice ID (SID)**. The external component is used for inter-domain routing. Its role is the same as that of IP prefixes in BGP, and we call the external part (*i.e.* ASN.geolocation.SID) a *GIRO prefix (G-prefix)*. The internal component consists of the subnet and host part, similar to that in the current IP address. The internal component (*i.e.* subnet and host) is used for routing inside the destination network, which is at the intra-domain level and not of interest for this paper. Figure 3 illustrates the GIRO address structure. Note that we do not present an exact address format in terms of how many bits each field has, since the focus of this paper is to evaluate the benefits of the general idea rather than spelling out all the design details. As a next step, we plan to investigate the details of the design, e.g., fitting the GIRO address structure into IPv6.

We now go into the details of the external component of GIRO address structure. A GIRO prefix is announced by its origin network into the Internet via BGP; routers keep routing table entries for GIRO prefixes, select paths to reach these GIRO prefixes, and GIRO prefixes can be aggregated in the routing table. The first field in a GIRO prefix is its AS number. In the case that the network does not have an AS number (*e.g.* it does not run BGP), its provider’s AS number will be used. If such a network has multiple providers, it may have multiple GIRO prefixes, one from each provider, as suggested by Shim6 [15]. Putting network ID in the first field of the address is a key difference between GIRO and previous geographic routing schemes. This ensures that packets are always routed to the correct destination networks, and appropriate ISP polices can be applied based on the network ID. The geographic information serves as a secondary hint in routing decisions, rather than the primary metric like in previous schemes.

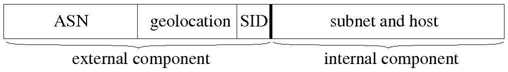
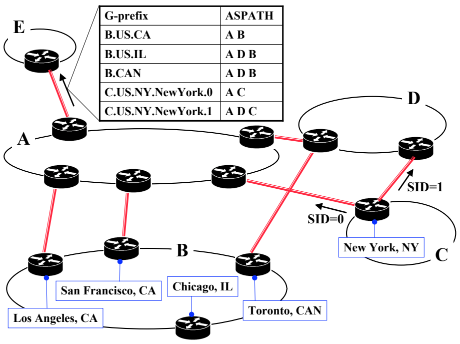


Fig. 3. GIRO address structure



G-prefix	ASPATH
B.US.CA	A B
B.US.IL	A D B
B.CAN	A D B
C.US.NY.NewYork.0	A C
C.US.NY.NewYork.1	A D C

Fig. 4. Example of GIRO address aggregation.

The second field in the GIRO prefix is its geographical location. One design question is how to represent and encode the geolocation in the address. The solution should allow easy calculation of geographic distance between addresses, and enable address aggregation at different levels. We decided to use longitude and latitude to represent geolocation and encode them in a way similar to the World Geographic Reference System (Georef) [4]. Take longitude encoding as an example. The first bit denotes whether the location is in the West hemisphere ($[-180°, 0°]$) or East hemisphere ($[0°, +180°]$), the second bit denotes whether the location is in the West or East half of its hemisphere, and so on. For instance, the encoding of the first two bits of longitude are: 00 for $[-180°, -90°]$, 01 for $[-90°, 0°]$, 10 for $[0°, +90°]$, and 11 for $[+90°, +180°]$. Using more bits will be able to represent the geolocation in finer resolution. This encoding scheme satisfies our requirement of distance calculation and aggregatability. For the ease of presentation, in the rest of the paper, we present the geolocation in the form of *country.region.city*.

The third field of a GIRO address is a "traffic slice ID," to facilitate traffic engineering of multi-homed networks. The idea is that, for a multi-homed AS, it uses the same SID for all the prefixes that originate from the same geolocation and are served by the same provider. This approach creates a finer-grain control over incoming traffic, since remote routers in the network may maintain different paths to reach different prefixes, even though the prefixes originate from the same geolocation. This is the case of the New York router in AS C in Figure 4 that announces SID 0 to AS A and SID 1 to D, thus allowing AS A to have two different paths to reach the destination prefixes. This is shown in the routing table of A's represented in the figure, which has two entries (*C.US.NY.NewYork.0* and *C.US.NY.NewYork.1*) to reach the NY router of AS C.

The new address format also brings new operational issues. For instance, there can be cases where it's necessary to migrate subnets to different SIDs, *e.g.* to adapt to dynamic traffic demands. In this case each host needs to be remapped to a different SID. The mapping can be published in DNS or pushed to a communication layer at the end host as in shim6 [15]. GIRO does not support anycast routing as it is in the current Internet routing, since every GIRO prefix has its geolocation encoded. To implement anycast, the hosts need to map the destination to a particular GIRO prefix and use it to communicate. Adding support for anycast routing is part of our planned future work.

B. Route Aggregation

Adding geographic information into the address structure also opens the door to better route aggregation. With GIRO address structure, route aggregation can be done at various level: ASN, geolocation, and SID. We assume that routing table entries can be aggregated if their prefixes are continuous (thus can be represented by a single shorter prefix) and their AS-level paths are the same. For instance, in Figure 4 the city level entries *B.US.CA.LA* and *B.US.CA.SF* are aggregated in a single entry *B.US.CA*, that is propagated from A to E. Another example, if the AS paths to reach all the prefixes originated by the same AS are the same, then they can be aggregated at the AS level. This greatly enhances the aggregatability of prefixes and should lead to smaller global routing table size.

C. Geographic Information in Routing Announcements

We add geographic information to routes by attaching the geographic distance of each AS hop to routing announcements. For instance, in Figure 5, the AS path [A B C] goes through three ASes via ingress and egress routers of each AS. Each border router, based on the geolocation information encoded in router addresses, will calculate the geographic distance from its previous egress/ingress router, and attach the distance in the BGP announcements. Based on the per-hop distances, a router can calculate its end-to-end distance to the destination GIRO prefix and use it to improve its route selection.

D. Path Selection

In order to select the shortest geographic paths, we replace the BGP hop count comparison with a geographic distance comparison, as shown in Table I, step 2. Note that each router is able to compute the geographical length of each route since GIRO explicitly includes this information in route announcements. However, since

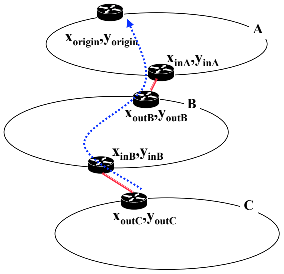


Fig. 5. Adding geographic information to routes.

Step	Description
1.	Highest LocalPref
2.	**Shortest geographic distance with resolution δ**
3.	Lowest AS hop count
4.	Lowest origin type
5.	***Late-exit* policy ⇒ lowest MED** ***Shortest-path* policy ⇒ shortest geographic distance**
6.	eBGP-learned routes over iBGP-learned
7.	Route with lowest IGP distance
8.	**Shortest geographic distance**
9.	Lowest router ID

TABLE I

GIRO DECISION PROCESS. CHANGES INTRODUCED BY GIRO ARE IN BOLD TEXT; THE REST IS THE SAME AS IN BGP DECISION PROCESS.

distance is measured in miles, step 2 in Table I might often be too selective and prevent the evaluation of routes that have slightly lower distance, but are actually better in terms of end-to-end performance or local cost (*i.e.* IGP distance). Therefore, we introduce a new parameter δ that represents the resolution that geographic distances are measured in step 2. The value of δ is an operational parameter configured by each ISP. If δ is small, the decision process is essentially optimizing the global cost of the route by minimizing the end-to-end distance to the destination. On the other hand, if δ is large, step 2 becomes less selective and the decision process essentially optimizes the local cost by *e.g.* applying early-exit in step 7. Therefore, the parameter δ is a knob that allows each ISP to tune the trade-off between optimizing the global cost and optimizing the local cost of each route. Note that we place the AS hop count comparison in step 3, since depending on the value of δ, step 2 can still output several candidate routes with different AS paths. From these routes, step 3 selects the ones with the shortest AS hop count.

Besides step 2, we introduce another step in the decision process, step 8, that selects the route with the shortest geographic distance. This step is only applied when all parameters of the remaining routes are equivalent.

E. Egress Point Selection

Given a set of equal-preference and equal-hop-count routes, the default BGP policy - *early-exit* - always picks the route that passes through the closest exit-point. Early-exit is a greedy policy since it only optimizes the local cost without regarding the global cost of the selection. Previous work[12] shows that the global cost of routes selected by early-exit is suboptimal, in the sense these routes usually are not in the shortest path to the destination. In this section we investigate an alternative policy to early-exit called *shortest-path*, that selects the exit point that is in the shortest-path to the destination. Even though shortest-path provides the optimal solution in terms of global cost, we need to assess the sacrifice in terms of local cost, *i.e.* measure the cost of carrying a route longer inside each individual AS and compare it to that in the early-exit case.

Figure 6 shows an example of two neighbor domains A and B that interconnect at three different points. Links are labeled as $g|w$, where g represents the geographical distance and w represent the IGP weights. The labels in the inter-domain links ($R_1 - R_4$, $R_2 - R_5$ and $R_3 - R_6$) represent physical distances. Suppose R_0 wants to find the minimum cost path to reach R_7. In order to do this, R_0 needs to combine the weights of the path segments in B with the weights of the path segments in A. Even if A's weights are announced through MEDs in BGP[2], it is unclear how to combine them with B's local weights, since different domains may use different scales for their internal metrics. Furthermore, the weights of the inter-domain links are unknown, even though R_0 knows their physical length from *e.g.* inspecting the location information embedded in the announced routes. Therefore, instead of using weights, a more straightforward alternative to compute the shortest path from R_0 to R_7 is to use geographical distances[3].

We consider three different policies for egress point selection:

- **Early-exit**: the default BGP policy. In absence of other information, BGP selects the closest exit point in terms of IGP weight (leftmost route in Figure 6).
- **Late-exit**: also known as *cold-potato*, selects the exit point that minimizes the cost inside the neighbor domain (rightmost route in Figure 6). Late-exit requires some level of cooperation between the ISPs, *i.e.* usually the ISP performing late-exit is a provider that is bounded by contract to deliver traffic to the customer while minimizing the costumer's local cost. The late-exit policy is implemented in BGP by the MED mechanism, *i.e.* within the same AS path, the route with the lowest MED is selected.

[2] In fact MED values are often related to IGP weigths[13].

[3] In some networks, IGP weights are a direct conversion of geographical distances between routers[1].

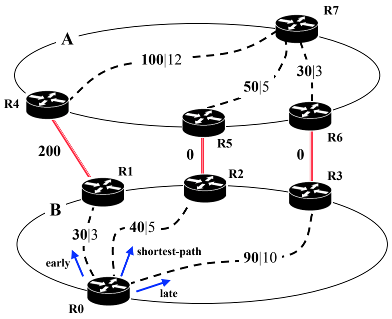


Fig. 6. Routing policies.

Policy	Description	Metric
Early-exit	select closest IGP exit	IGP weight
Late-exit	select exit such that link cost of neighbor is minimized	MED
Shortest-path	select exit on the shortest path to destination	geographic distance

TABLE II
INTRA-DOMAIN ROUTING POLICIES.

- **Shortest-path**: the exit point is selected so it minimizes the physical distance to the destination. In Figure 6 this is represented by the middle route. This policy may lead to a win-win scenario if the local cost of both ISPs is reduced. For instance, in Figure 6, both A and B reduce their local costs if traffic from R_0 to R_7 *and* traffic from R_7 to R_0 is carried over the shortest path. The shortest-path scenario yields a local cost of 5+5=10 for both A and B. In contrast, if both A and B were applying early-exit, the costs would be 12+3=15 for A and 10+3=13 for B.

In order to accommodate different policies, we include step 5 in the decision process of Table I. Late-exit and shortest-path are implemented in step 5, while early-exit is implemented in step 7. Step 5 selects an exit point for each AS path, consistent with the policy established with the neighbor domain [4]. Table II lists the possible policies and the column "Metric" indicates the cost metric that each policy tries to minimize.

IV. Evaluation

A. Inter-domain Route Selection

We simulate the decision process of Table I using a PoP level topology extracted from Rocketfuel[21]. The Rocketfuel-measured topology consists of 668 AS level links between 67 ISPs mostly from tier-1 and other large ISPs connected directly to tier-1. If not in number, these ISPs are representative in terms of originated prefixes[21]. Each ISP is mapped to a set of PoPs annotated with their geographical location in the form of latitude/longitude. Inside an ISP, each link between a pair of PoPs is annotated with its inferred IGP weight. We run Dijkstra algorithm to compute the total cost between border PoPs, *i.e.* PoPs that were connecting to other ISPs. Relationships between different autonomous systems were inferred using the PTE algorithm[9], and are classified in two types: peer-to-peer and customer-to-provider. We simulated step 1 in Table I by preferring customer routes over peer and provider routes, and preferring peer routes over provider routes. We computed paths between ISPs using the no-valley rule, *i.e.* an ISP does not do transit between providers or peers.

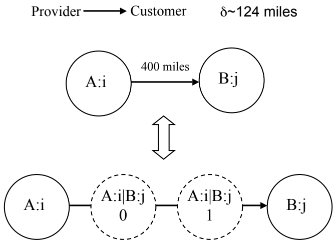


Fig. 7. Modeling the distance between PoPs.

In order to simulate BGP paths, we abstract the decision process into the following steps: (1) local preference based on policy, (2) AS path length and (3) random tie-breaker. We convert these AS level paths to PoP level paths by assuming the default early-exit policy. In a path A–B–C we randomly pick a PoP in A and C, and starting from A, we apply early-exit until we reach the origin in C.

We compute GIRO paths in a different way. We first build a PoP level topology and insert *virtual nodes* between PoPs according to the geographical distance between them, as shown in Figure 7. ISPs A and B are connected at PoPs $A\!:\!i$ and $B\!:\!j$, where i and j are PoP identifiers used inside each ISP. We configure the δ in Table I to 1ms in latency, *i.e.* $\delta = \frac{\frac{2}{3}c}{1ms} \simeq 124$ miles, where c is the speed of light in vacuum, and $\frac{2}{3}c$ is the speed of light in fiber optic[17][5]. The distance between the PoPs $A\!:\!i$ and $B\!:\!j$ is 400 miles, but using resolution of $\delta \sim 124$ miles, we convert it to a distance of 3 units. Therefore, we insert two virtual nodes between $A\!:\!i$ and $B : j$ to account for this distance, as shown in Figure 7. We finally feed this topology to the path computation

[4] These policies can be marked in routes *e.g.* through the use of special community values.

[5] This speed is actually close to the speed extracted from the slope of Figure 2, but slightly higher because of delays related to transmission and queuing.

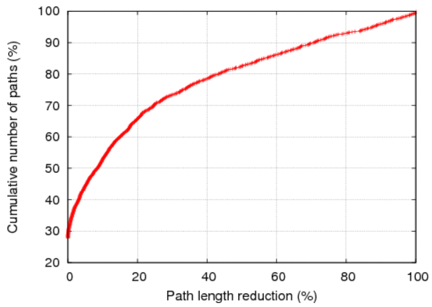


Fig. 8. GIRO path length reduction compared to BGP.

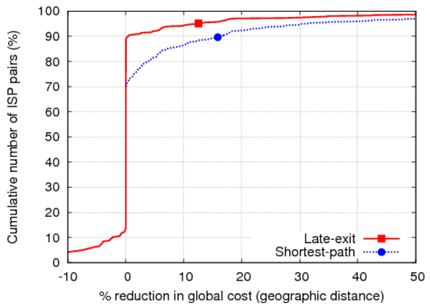


Fig. 9. Global cost reduction compared to early-exit.

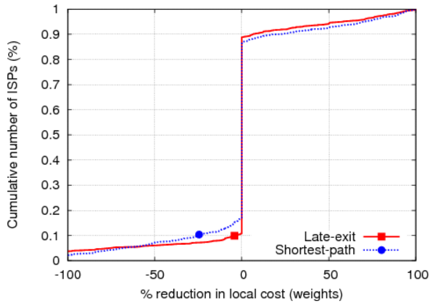


Fig. 10. Local cost reduction compared to early-exit.

algorithm we used for BGP, obtaining the set of PoP level paths traversed by GIRO routes.

Figure 8 shows the path length reduction achieved by GIRO. About 20% of GIRO paths have a length reduction of more than 40% compared to BGP. The case in Figure 1 showed up in the simulation results, where the difference in end-to-end path length between BGP and GIRO is more than 2,800 miles.

B. Egress Policy Evaluation

In this section we evaluate the policies described in Table II. We are mainly interested in assessing the sacrifice of local cost when applying shortest-path policy. We use the same Rocketfuel PoP level topology described previously, but this time we only look at paths between neighbor domains. For each pair of neighbor ISPs, we randomly select a PoP in each ISP that is not directly connected to the neighbor. The reason is to have scenarios where we can evaluate the dependency between the costs and internal weights of each ISP, which would not be possible if origin and destination PoPs were directly connected. In an ISP pair A–B, we apply the same policy on the flow $A \rightarrow B$ and $B \rightarrow A$ and compute the local cost and global cost of the scenario assuming both A and B apply the same policy towards each other. For instance, in Figure 6, the total local cost of B when applying shortest-path is given by $w_{inB} + w_{outB} = 5 + 5 = 10$, where w_{inB} is the weight of the link traversed by the incoming flow from R_2 to R_0 and w_{outB} is the weight of the link traversed by the outgoing flow from R_0 to R_2. We define global cost as the total length of the path traversed by the incoming and outgoing flows. In Figure 6, the global cost of shortest-path is given by the distance of the route from R_0 to R_7 and the route from R_7 to R_0, yielding 90+90=180 miles.

Figure 9 shows the global cost reduction of each policy when compared to the default early-exit for all pairs of neighbor ISPs. We observe that late-exit is the policy that achieves lower global cost reduction. In fact the global cost of late-exit would be the same of early-exit if all connections between ISPs were bidirectional. However, since this is not always the case, 80% of cases have the same cost as early-exit, but there are 20% of cases evenly divided in lower cost and higher cost when compared to early-exit. Shortest-path achieves the greatest global cost reduction. About 30% of cases correspond to shorter paths when compared to early-exit. Note that we have been comparing paths assuming the amount of traffic carried in each one is similar. In reality we might want to give more importance to paths that carry more traffic over the ones that carry a smaller amount of traffic [12], which is part of future work.

Having look at the global cost, we now look into how the local cost changes compared to early-exit. Figure 10 shows the quantitative change in local cost for each ISP for late-exit and shortest-path. For both policies, less than 18% of paths cause an increase in local cost (negative part of x-axis), and about 12% of paths cause a decrease.

We perform a win-lose analysis for each pair of ISPs relative to the case where they route traffic to each other based on early exit. For instance, in the pair of ISPs A–B, if both A and B see their local cost reduced compared to early-exit, we say they achieve a *win-win* situation. If their local cost remains the same, then it's a *same-same* situation, and so on. Figure 11 shows the fraction of each case for each of the policies. We observe that shortest-path has mostly outcomes in same-same and win-lose cases. Note that in a win-lose scenario, the same ISP may sometimes lose and other times win, but

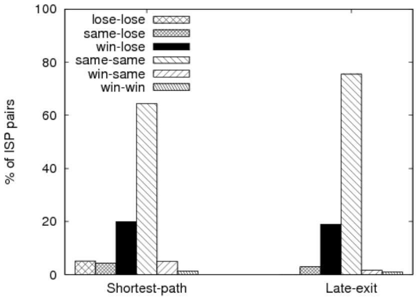


Fig. 11. Win/lose analysis for ISP pairs.

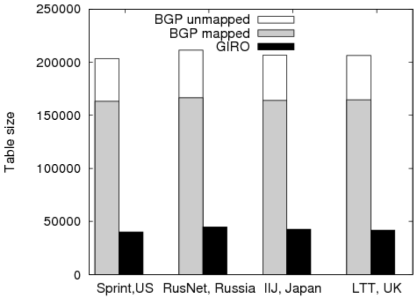


Fig. 12. Estimation of GIRO table size.

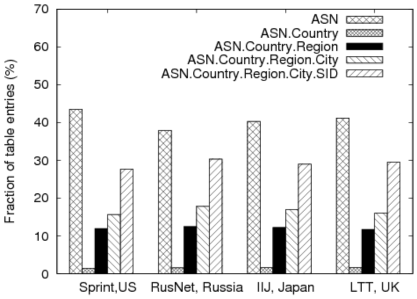


Fig. 13. Decomposition of GIRO table in different aggregates.

there is at least one ISP winning. However, in terms of global cost, all ISPs win as indicated in Figure 9. Therefore, we believe the advantage of using shortest-path in achieving the optimal global cost overcomes the sacrifice of increasing the local cost for some paths, which is limited as indicated by Figures 10 and 11.

C. Geographical Aggregation

To evaluate GIRO aggregation, we use BGP tables extracted from RouteViews[16] and RIPE[3] from January 2007 to March 2007. The union of these tables contained 246,547 prefixes announced from 24,605 different autonomous systems. This set do not include multiple origin prefixes, *i.e.* prefixes that are announced from different ASes. The reason is that the mapping to locations for these prefixes is ambiguous, since they can be announced from very distant places[6]. We find the geographical locations of prefixes using GeoLite City database from Maxmind[2], which maps each IP address to a country, region and city. For each prefix in our data set, we randomly pick three IP addresses (excluding .0 and .255) and map these addresses to locations using GeoLite. We then map each prefix to a location by taking majority voting on the previous three mappings. We are able to map 195,992 prefixes using this method, which corresponds to about 80% of the prefixes in our set. We note however that the mapping from prefixes to locations is usually not one-to-one, *i.e.* a same prefix may be dispersed across different locations, as reported by[8]. However, for most prefixes (/24 and longer) the one-to-one mapping is a reasonable assumption.

Since GIRO routing table depends on a router's location, we use four routers in distinct geographic locations: US, Russia, Japan and United Kingdom. We convert each prefix in these routers' routing table to a G-prefix and aggregate different G-prefixes using the following rules. First, we do not aggregate G-prefixes that have distinct AS paths. The reason is that in order to make a fair comparison between GIRO and BGP, we want to keep the path diversity of BGP, at least at the AS level. Therefore, we assign different traffic slice(SID) numbers for each AS path announced from (ASN,location) pairs, as *e.g.* the NY PoP in Figure 4. Second, we aggregate geo-prefixes to the shortest possible form, while keeping the consistency of the forwarding state. For instance, in Figure 4, the entries from Los Angeles and San Francisco were aggregated in a single entry $B.US.CA$. If these two entries were aggregated at the country level $B.US$, then the forwarding state would be compromised since there would be two next-hop alternatives for destinations in Chicago (B and D) , even though only one of them is valid (B) [7].

Figure 12 shows the comparison of table size between GIRO and BGP after applying the above aggregation rules. The "BGP unmapped" entries represent prefixes that we could not be map to locations. We observe that GIRO table size is about 25% of the mapped BGP entries. We now investigate how much of this reduction is due to geographical aggregation. Note that all G-prefixes except the short form ASN are a result of geographical aggregation. Figure 13 indicates that about 40% of entries are in the fom ASN, which means that 60% of entries resulted from some form of geographical

[6] We found these cases to represent a very small percentage of total prefixes.

[7] In this example we assume *any* match instead of *longest* match, even though in practice forwarding is done based on longest matching.

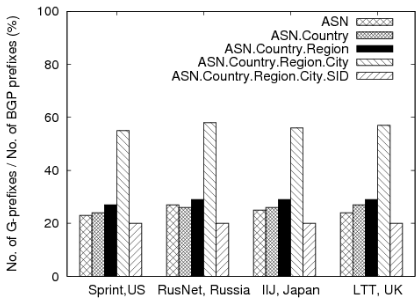


Fig. 14. Number of G-prefixes over number of BGP prefixes for different aggregates.

aggregation. However, these different aggregates were not originated from the same number of prefixes. Figure 14 shows the ratio of number of G-prefixes over number of prefixes for each aggregate. We observe that all aggregates achieve a similar level of compression close to 25% except for entries of type $ASN.country.region.city$ that only compress between 55% and 58%. We believe the reason is that these cases usually correspond to single-homed ASes that do not originate many prefixes as the multi-homed ones.

V. Discussion

In this section we describe some open issues and some new functionalities of GIRO that deserve further attention. As described in section III, GIRO border routers attach to announced routes the geographic distance traversed in each AS. An alternative to this approach would be each border router to stamp its absolute location in the announcements in the form of latitude/longitude. There are advantages and disadvantages in these two approaches.

Having the absolute location in routes may help in doing better fault diagnosis. For instance, ISPs could do geographic circumvention of network instabilities. This is potentially useful in cases of persistent instabilities restricted to a geographic area, *e.g.* the recent Taiwan earthquake[25]. In these cases, routes may flap several times and create periods where data paths are not available. ISPs may respond to these situations by explicitly avoiding routes that traverse the problem area. The absolute geographical location might also help in detecting source address spoofing. For instance, if border routers also stamp their geographical location in data packets, then the destination may detect source address spoofing by verifying if the packet came from the same location where the route to the source was announced.

The disadvantage of using absolute location information is that it may rise privacy concerns, since some ISPs may not be willing to disclose the structure of their networks, namely the locations of their geographical points of presence.

An additional benefit of GIRO we did not discuss regards prefix hijacking. Note that since now the ownership of each G-prefix is implicit in the address (by including the ASN), it is no longer possible to forge prefix ownership. However, it is still possible to forge routing announcements by including false AS links. For instance B may attack C by advertising false link B-C to A. There are ways for A to mitigate these attacks. For instance, A can check if the location information associated with C is too far away from previous C's announcements. An alternative way is to temporarily accept the route under test and measure its round-trip time. If this time is much different from the one derived from the geographic distance, then don't accept it or give it lower preference.

Regarding the incremental deployment of GIRO, note that some of the features may actually be implemented in IPv4 through the use of BGP communitites. For instance, similarly to what was proposed in [14],the location of the origin, as well as the geographical distances traversed by the route could be encoded in BGP communities. Alternatively, GIRO addresses would fit the 128 bits of an IPv6 address.

VI. Related Work

Previous work addressing the relation of geography and inter-domain routing can be divided into analysis of geographic properties of Internet routes[22], [20], [8], and design of addressing schemes that include some form of geographical information[7], [11], [10]. Subramanian *et al.* [22] analyze the geographic properties of Internet routes to find that circuitousness of paths, *i.e.* the ratio between end-to-end real distance and direct distance, tends to be greater when paths traverse multiple ISPs. Spring *et al.* [20] do a first systematic measurement of path inflation in the Internet, and point some causes that aggravate the problem, namely the hop count tie-breaker in the BGP decision process. The first work to study the locality of prefixes is the one by Freedman *et al.* [8]. They find that about 65% of discontinuous prefixes announced from same (AS,location) pair comes from discontinuous allocations. Both Francis[7] and Huston[11] compare two different types of addressing schemes for Internet routing: provider-based and geographical. Provider-based addressing is similar to current Internet addressing, where providers delegate chunk of addresses to their customers. Geographical addressing follows a geographical hierarchy, similar to the portion country.region.city of G-prefixes. For geographical addressing, they assume that routing is still done based on geography, and therefore ISPs need to be interconnected in mesh at each location. Hain[10] proposes an address scheme where each host is given an address based on its latitude and longitude, and there is still the premise that routing is done primarily by geography. To the best of our knowledge, our work is the first that uses a form of geographical addressing while keeping policy routing between ISPs, not requiring any change to Internet topology.

Teixeira *et al.* [24] propose TIE, a scheme that replaces hot-potato routing with a more generalized metric $w' = \alpha \cdot w + \beta$, where w is the IGP distance, and α and β are configurable values to tune the selection of egress points. When computing α and β, their scheme uses a delay threshold to decide the change to a different egress point. However, if route geographic distance is available, the delay threshold can be replaced by a distance threshold, much easier to compute.

Francis *et al.* [26] describe a scheme that decouples address hierarchy and physical topology by using tunneling over an overlay with virtual prefixes. Their scheme achieves a compression of the routing table by more than one order of magnitude, but at the cost of increasing the length of routing paths. Furthermore, its not clear how policy routing between ISPs is maintained in their scheme. In contrast, GIRO achieves a significant reduction of table size, while effectively reducing the length of routes.

VII. Conclusion

In this paper we proposed a new network routing and address scheme, GIRO, which incorporate geographic location information to assist routing decisions. Our solution departs from previous geographical addressing proposals by putting the AS information as the most significant bits in the address, and putting the geographical information only after the AS information. This address structure supports routing policies while provides geographical information to the routing decision process. When everything else is equal among multiple available paths, GIRO improves over current route selection mechanism by selecting paths with shorter geographic distances. Our simulation results using a Rocketfuel-measured Internet topology show that, within the constraints of routing policies, GIRO can both significantly shorten the routing paths through the selections of shorter AS paths and better neighbor domain exit points, and effectively reduce the global routing table size through geographical routing aggregation.

References

[1] Internet2 NOC. `http://www.abilene.iu.edu/i2network/maps--documentation.html`.

[2] MaxMind GeoLite City. http://www.maxmind.com/app/geolitecity.

[3] Regional Internet Registry Data. ftp://www.ripe.net/pub/stats.

[4] P. Dana. Coordinate Systems Overview. http://www.colorado.edu/geography/gcraft/notes/coordsys/coordsys.html.

[5] S. Deering. Metro-Based Addressing: A Proposed Addressing Scheme for the IPv6 Internet. Presentation, Xerox PARC, July 1995.

[6] G. Finn. Routing and Addressing Problems in Large Metropolitan-Scale Internetworks. Technical Report ISI-TR-2001-552, Information Sciences Institute (ISI), March 1987.

[7] P. Francis. Comparison of geographical and provider-rooted internet addressing. In *JENC5: Selected papers of the annual conference on Internet Society/5th joint European networking conference*, pages 437–448, 1994.

[8] M. Freedman, M. Vutukuru, N. Feamster, and H. Balakrishnan. Geographic Locality of IP Prefixes. In *Internet Measurement Conference (IMC) 2005*, Berkeley, CA, October 2005.

[9] L. Gao. On inferring Autonomous System Relationships in the Internet. In *IEEE/ACM Transactions on Networking*, volume 9, pages 733–745, 2001.

[10] T. Hain. Application and Use of the IPv6 Provider Independent Global Unicast Address Format. *Internet Draft*, August 2006. draft-hain-ipv6-pi-addr-use-10.txt.

[11] G. Huston. IP Addressing Schemes - A Comparison of Geographic and Provider-based IP Address Schemes. December 2004. http://ispcolumn.isoc.org/2004-12/addressing.pdf.

[12] R. Mahajan, D. Wetherall, and T. Anderson. Mutually Controlled Routing with Independent ISPs. In *USENIX NSDI*, April 2007.

[13] D. McPherson and K. Patel. Experience with the BGP-4 Protocol. *RFC 4277*, January 2006.

[14] D. Meyer. BGP communities for data collection. *Request for Comment (RFC): 4384*, 2006.

[15] E. Nordmark and M. Bagnulo. Level 3 multihoming shim protocol. *Internet Draft*, November 2006. draft-ietf-shim6-proto-07.txt.

[16] U. of Oregon. RouteViews Routing Table Archive.

[17] R. Percacci and A. Vespignani. Scale-free behavior of the internet global performance. *European Physical Journal B 32*, pages 411–414, 2003.

[18] Y. Rekhter, T. Li, and S. Hares. Border Gateway Protocol 4. RFC 4271, Internet Engineering Task Force, January 2006.

[19] S. Savage, A. Collins, E. Hoffman, J. Snell, and T. Anderson. The end-to-end effects of internet path selection. In *SIGCOMM '99: Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication*, pages 289–299, New York, NY, USA, 1999. ACM Press.

[20] N. Spring, R. Mahajan, and T. Anderson. Quantifying the causes of path inflation. In *ACM SIGCOMM*, 2003.

[21] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson. Measuring ISP Topologies with Rocketfuel. *IEEE/ACM Trans. Netw.*, 12(1):2–16, 2004.

[22] L. Subramanian, V. N. Padmanabhan, and R. H. Katz. Geographic properties of internet routing. In *Proceedings of the General Track: 2002 USENIX Annual Technical Conference*, pages 243–259, 2002.

[23] H. Tangmunarunkit, R. Govindan, and S. Shenker. Internet Path Inflation Due to Policy Routing. In *SPIE ITCom*, 2001.

[24] R. Teixeira, T. G. Griffin, M. G. C. Resende, and J. Rexford. Tie breaking: tunable interdomain egress selection. In *CoNEXT'05: Proceedings of the 2005 ACM conference on Emerging network experiment and technology*, pages 93–104, New York, NY, USA, 2005. ACM Press.

[25] T. Underwood and A. Popescu. Quaking Tables: The Taiwan Earthquakes and the Internet Routing Table. *Nanog 39*, February 2007.

[26] X. Zhang, P. Francis, J. Wang, and K. Yoshida. Scaling Global IP Routing with the Core Router-Integrated Overlay. In *the 14th IEEE International Conference on Network Protocols*.