University of Oslo
Department of Informatics

Topology Construction for Inter-Domain Network Protocol Simulations

Tarik Čičić,
Stein Gjessing,
Øivind Kure

Research Report Nr. 296

ISBN 82-7368-246-3
ISSN 0806-3036

30th August 2001

Abstract

Inter-domain network protocol simulations are often resource-demanding. A reduction of the inter-domain topology size can make the simulations more tractable. However, the simulations must yield results that are applicable to the real system.

In this report, we describe a procedure for selecting a topology suitable for simulations of global, inter-domain protocols in the Internet. The procedure preserves the significant properties of the Internet autonomous system topology. The resulting topology is a subset of this real inter-domain topology. Both topologies have similar graph properties. The shortest path route between any two domains in the resulting topology is identical to the shortest path between these two domains in the real topology.

We also propose an inter-domain delay model based on the geographic location of the domains. Though simple, this model provides a close approximation to the real inter-domain communication delay.

Keywords:

Inter-domain, protocol, topology, latency, simulation, validation.

1 Introduction

In the process of inter-domain protocol development, the inherent complexity of a real test environment, as well as the complexity of the protocols themselves, advocate use of simulation as a development tool. The simulation detail level will set limits on the manageable topology size. On the other hand, the simulation must yield valid results — it must be representative for the real system.

In this report we describe our experiences with selecting a topology that can be successfully used in simulations of global, inter-domain protocols. We show how a topology of limited size, but with properties similar to the real Internet inter-domain topology, can be derived from publicly available topology information. The resulting topology has a lower number of domains, while its density and average path length are similar to those of the Internet inter-domain topology. In addition, we propose a simple inter-domain delay model based on available geographic data, which is sufficiently accurate for e.g. protocol performance comparison purposes.

Our work is motivated by the necessity of topology selection for our project on cost and performance evaluation of two inter-domain multicast routing protocols[1]. The methodology and results presented in this report are independent of the routing paradigm and protocol, and hence relevant for any inter-domain protocol simulation.

This report is organized as follows. In Sec. 2 we provide background information on the structure of the Internet and the options for the simulation topology choice. The properties of a suitable target topology are specified in Sec. 3 as a list of requirements the target topology must meet. Section 4 describes the information sources for our work. In Sec. 5 we provide a step-by-step description of how the initial data has been transformed to a topology that conforms to our requirements. Section 6 validates the resulting topology by comparing it to the available knowledge about the real inter-domain topologies. In Sec. 7 we conclude the report and present ideas for future work.

2 Background

2.1 Hierarchical Structure of the Internet

The Internet can be coarsely regarded as a two-level hierarchical network, where the upper level is a topology of interconnected autonomous systems (AS)[2], and the lower level represents the AS networks themselves. Each such network is autonomous in sense of local policies, such as routing, access control and traffic policies. Different domains (ASes) in the Internet have widely different sizes and internal structure. Each domain (AS) in the Internet has its unique AS number.

The inter-domain connectivity is maintained through use of an inter-domain routing protocol. The most widely used inter-domain protocol in the Internet today is the "Border Gateway Protocol" (BGP-4, [7]). Domains are interconnec-

[1] The Multicast Source Discovery Protocol, MSDP [5] and the Border Gateway Multicast Protocol, BGMP [4].

[2] Terms "autonomous system" (AS) and "domain" are not synonymous in general. In our work we assume that one AS represents one network domain, and use the terms interchangeably.

Date	Number ASes	Number Peerings
Feb. 1998	3464	7722
Feb. 1999	4767	8982
Feb. 2000	6978	13858
Feb. 2001	10359	21023

Table 1: Growth of the Internet in recent years. The data is based on NLANR traces (Sec. 4).

ted using point-to-point BGP peerings between routers called border gateways. Each BGP router announces the reachability of networks within the local domain to its external peers. It also maintains a coherent view of the reachability of networks in other domains with its internal peers. Finally, each domain can offer its transport services to other domains by announcing that a given network in another domain can be reached through it. Today many routers also support "Multiprotocol Extensions for BGP" (MBGP, [1]), which is needed for emerging inter-domain network services such as multicast.

Each border gateway maintains a routing table that, for each destination network, includes next-hop information, path (a list of ASes to be visited on the way to the destination network) and several other fields. When more than one route to a destination exist, the gateway selects one "best" route using a domain-internal set of rules (routing policy). Only the best routes are announced to the neighbors. Also, the gateway must aggregate the address information for networks accessible through it, if possible, to minimize the amount of routing state.

The number of ASes in the Internet has been on rise in recent years (Tab. 1). This inter-domain topology evolves by increasing the number of domains and inter-domain links. Also, domains usually persist for a long period after creation. We have registered that less than 10% of the AS numbers present in the BGP AS number set from one year were not present the next year (for years 1998, 1999 and 2000).

2.2 Topology Types

In communication network simulations, we distinguish between random (synthetic) topologies and real network topologies.

Random topologies are increasingly popular. It is known that modern topology generation algorithms construct random networks similar to real Internet topologies [10]. Also, the model parameters can be used to cover a range of possible real situations, which makes the random topologies suitable for general simulation studies. Simulations that use random topologies are usually performed on a class of similar networks (i.e. with same size, density etc.) in order to detect possible parameter correlations and tighten the statistical confidence intervals. This, however, implies increased simulation time and computing resource cost.

Real topologies have often been used in simulations. Typically, the topology of e.g. a research network would be coded in a format suitable for use by a simulator. Use of real topologies makes the simulation process more efficient, since a single (or few) fixed, relevant topologies are used as the basis for the simulation. Also, the simulation results from a real topology can be validated by comparing with (implemented) components of the real system.

3 Target Topology Properties

Our target topology and its associated delay model must meet the following requirements:

R1 the network parameters such as the inter-domain network node degree and diameter must be similar to the real global inter-AS topology.

R2 adopt the real Internet topology layout as far as possible.

R3 provide communication delay in the same size order as what is experienced in the Internet

R4 be manageable for the available simulation hardware and software.

In our inter-domain protocol studies, we model the topology as a graph where the vertex represent domains and the edges represent the inter-domain links. The graph properties are known to influence the simulation results in most simulation scenarios. Requirement (R1) specifies that the target topology must be similar to the Internet from the graph-theoretic point of view.

It is possible to deduce the global Internet topology from the publicly available BGP traces, and thus fully satisfy (R1). But, the full inter-AS topology counts more than 10000 domains. Depending on the simulation detail level, such a large topology is probably too resource-demanding for most simulation systems. As an example, our simulation system (*ns*, [9]) can successfully simulate networks of some 1000 nodes on a platform with 1 GB RAM and ~1 GHz Pentium III Xeon processor[3]. Therefore, (R1) and (R4) cannot be satisfied fully and simultaneously. We need to decrease the total number of nodes and links in the topology.

When trading simulation time for topology size, we take care to keep the network attributes of interest for the routing algorithms unaltered (R2). In practice this means that the shortest path between domain pairs in the target and the real topology should be the same. This requirement would be difficult to meet using random domain and link selection, since the resulting paths would likely be different in the real and the target topology. We therefore apply deterministic algorithm operations only in our topology reduction procedure.

Communication delay is an important metric in many simulation scenarios. In the Internet, it depends on a range of parameters, including the topology, routing algorithm, network technology, physical location of components and traffic patterns. When simulating smaller network systems, the communication delay is often simulated by modeling the propagation delay of network links, packet transmission delay, queuing delays, node processing delay etc. When simulating global inter-domain communications, this level of detail is not possible to achieve. Each transit domain is simulated by a limited number of nodes, and the delay focus moves to parameters such as distances and ingress data processing.

[3]On the needed detail level, a 1000-node simulation consumes over 900 MB RAM. A further increase of network size leads to a disproportional increase in simulation time due to heavy use of disk caching.

While obtaining correct delay values in a large scale inter-domain simulation is clearly unrealistic, operating with close-to-reality delays is useful for protocol comparison under otherwise same conditions (R3).

4 Information Sources

There are several research projects engaged in collecting and analyzing the global Internet topology and traffic information, as well as many informative WWW sites. We describe the information sources we have used in our work, as well as several other related sources.

4.1 NLANR

The "National Laboratory for Applied Network Research" [6] has as its primary goal to provide technical, engineering, and traffic analysis support of high performance research networks. NLANR works towards a better understanding of the global Internet services and metrics. It collects and analyses data obtained through a range of passive and active measurements in scope of routing, data traffic etc.

The NLANR's BGP data service (http://moat.nlanr.net/AS/background.html) has played a major role in our work. It is formed as a public service that daily provides the global BGP routing state as seen from several transit gateways distributed worldwide. This data is freely available for public downloads.

4.2 CAIDA

The "Cooperative Association for Internet Data Analysis" [2] provides tools and presents information about the state of the global Internet infrastructure. It is an NLANR's offspring focused on the commercial market, but represents a substantial information source also for research communities. CAIDA collects, monitors, analyzes, and visualizes several forms of Internet traffic data concerning network topology, workload characterization, performance, routing, and multicast behavior. These analyses serve a variety of disciplines, including research, policy, education and visualization.

The CAIDA site describes a taxonomy of available research and visualization tools. CAIDA's MANTRA ("Monitor and Analysis of Traffic in Multicast Routers") is a tool for monitoring various aspects of multicast at the network level. The information can be depicted using the "Otter" visualization tool. MANTRA also provides multicast inter-domain connectivity maps, representing views of MBGP topology as seen from the individual routers[4]. Besides visualization of these maps, Otter also can present useful information like fan-out, route terminations and other multicast statistics.

CAIDA has developed the NetGeo system, which is a database and collection of Perl scripts used to map IP addresses, domain names and AS numbers to geographic locations. We have used NetGeo to determine geographic registration centers of autonomous systems.

[4]Six routers have been covered in the time of writing, belonging to FIXV (NASA), ORIX, Route-Views (Oregon IX), STARTAP, GIGABELL and DANTE projects.

4.3 Other Sources

We have used many other WWW resources in our quest for a realistic network topology. For instance, the hourly-updated BGP routing table statistics published at Telstra site [8], have been used in the verification process. The network latency statistics and current measurements can be obtained at WWW.UUNET.COM/NETWORK/LATENCY/, WWW.SPRINT.NET and WWW.MATRIX.NET, among others. The "CyberGeography" site (HTTP://WWW.CYBERGEOGRAPHY.ORG/) provides a range of links to the sites relevant for studies of the spatial nature of the Internet.

5 Topology Selection Procedure

In this section we describe the rationale and procedure for selecting a realistic inter-domain topology suited for global protocol simulations. The procedure complies with the requirements we have imposed on our target topology (Sec. 3). We present the initial data format as collected from our information sources (NLANR and CAIDA), and then, step by step, describe and discuss the procedure we have applied to "distill" the initial data into the target topology.

For improved understanding, we illustrate each part of the procedure by practical actions we have taken to create our target topology for inter-domain multicast simulations. These actions are described in four stages, and interleaved with the description.

5.1 Initial Information

We have used NLANR's BGP data service as the starting point for creating the simulation topology. This source has provided daily BGP routing state since November 1997. The router that collects the information, ROUTE-VIEWS.OREGON-IX.NET, stores views of the full routing table from each of the 35 transit gateways that participate in the project[5]. This data is published through a WWW interface. The amount of information in the daily traces is constantly increasing; in the time of writing the traces have reached 250 MB size and 3 million BGP routing entries. Each routing entry is written as a single text line (Tab. 2). It is not possible to determine which transport gateway has provided a specific line in the table.

The NLANR BGP data includes a huge amount of information. 35 gateways provide their routes to the NLANR trace, and there are more than 100 000 networks in the Internet. Despite the large data amount, the exact routing policies are not deducible from the trace — they are only implicitly contained in the selected "best" routes (indicated by the ">"-flag in the example in Tab. 2), and only for the 35 contributing gateways.

5.2 Topology Selection

Using all routes to select the inter-domain topology would significantly increase the graph density by links that are unused in practice. Using exact routing policies is not possible, as per-domain policies are unknown in general. We

[5]The number of gateways participating in the project has gradually increased from 20 to 35 today.

Flag	Network	Next Hop	Mt.	Wg.	Path
x	129.240.0.0/15	205.215.45.50		0	4006 1239 6453 8297 2603 224 i
x		195.211.29.254		0	5409 8297 2603 224 i
x		163.179.232.37		0	2551 1239 6453 8297 2603 224 i
x		216.140.14.127	91	0	6395 6453 8297 2603 224 i
x		165.87.32.5		0	2685 6453 8297 2603 224 e
x>		198.32.8.252		0	11537 2603 224 i

Table 2: BGP routing trace format, example. Six consecutive entries (out of 2.8 million) are presented. *Flag* 'x' marks the valid entries, '>' marks "best" entries. The destination *network* can be reached through *next hop* network, visiting domains specified in the *path*. *Metric* ("Mt.") is used in route calculation, *weight* ("Wg.")for breaking ties when multiple valid routes are available. 'i' or 'e' marks if the next hop is an internal or an external gateway.

therefore chose to select the inter-domain topology based only on the best routes in the trace. We believe that this, in combination with the shortest path routing in the simulator, will produce realistic routes in most cases.

As the first step towards our target topology, we construct the inter-domain connectivity graph $\mathcal{G}_0$. We parse the BGP data and extract all AS-number pairs $D1 \leftrightarrow D2$ occurring in any best route selected by any of the 35 gateways. For example, AS-path in the last line of Tab. 2 results in peering list $\{11537 \leftrightarrow 2603, 2603 \leftrightarrow 224\}$.

We have assumed bidirectional nature of all links in our topology. Strictly speaking, an AS path that includes "...$D1D2$..." says that data may flow from $D1$ to $D2$, but says nothing about $D2 \rightarrow D1$ flow. The assumption that the links are bidirectional is necessary, since the BGP information was collected by (only) 35 gateways, and $D2 \rightarrow D1$ may not occur in the traces even though it would be used in practice.

Stage I *On a single chosen day (21st February 2001) we collected the input data for our topology selection procedure:*

1. *global BGP routing state from NLANR*
2. *a set $\mathcal{S}$ of multicast ASes visible in AS 10764 (STARTAP) through MBGP data, which will serve as the base set for topology reduction (Stage II).*

We collected the data on a single day to minimize the amount of inconsistencies.

The BGP state (1) is stored in a ∼224 MB, 2.8 million lines text file, each line describing a BGP routing entry. The file has been parsed line-by-line to create the connectivity-pair list. Loop-back links $D \leftrightarrow D$ and possible double links were removed. The result of this first processing stage was a global topology $\mathcal{G}_0 = (V, E)$ consisting of $|V|$ =10068 domains and $|E|$ =13733 inter-domain links.

We note that the node number has decreased compared to Tab. 1. 218 domains were not part of any best route announcement. Further 73 nodes were organized in small clusters of less than ten domains disconnected from the rest of the topology, possibly due to routing update process going in parallel with

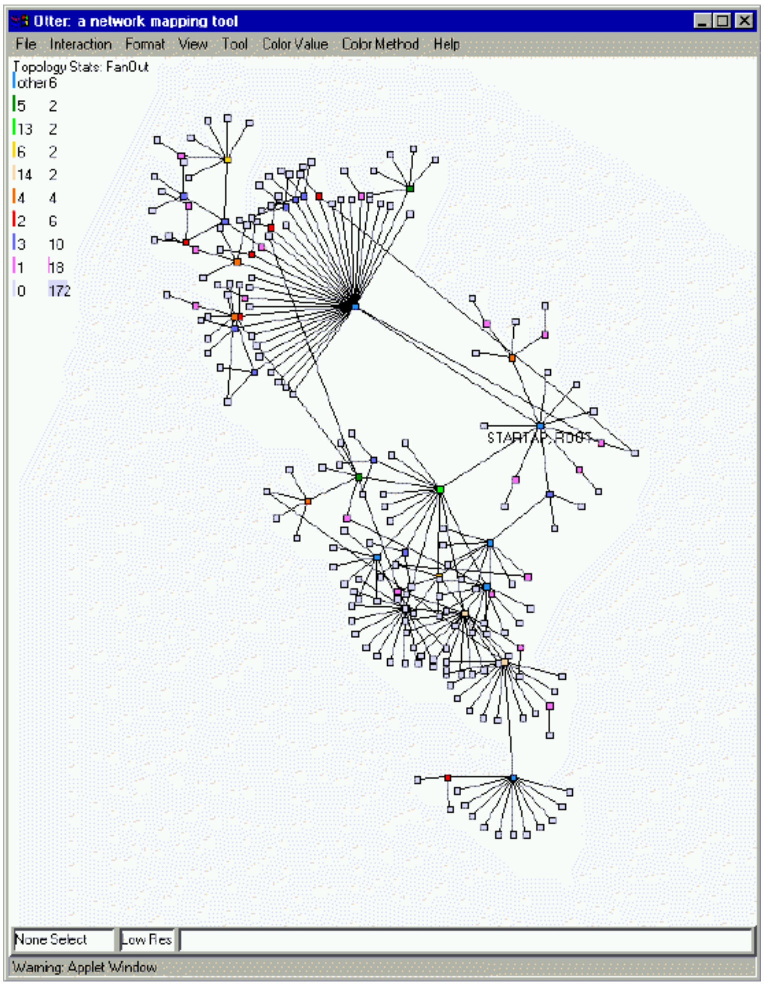

Figure 1: MBGP multicast peerings as seen in STARTAP on 21. February 2001

the NLANR data collecting. These 291 nodes are excluded from the topology selection procedure input.

Set $\mathcal{S}$ (2) was created using CAIDA's Otter visualization tool (snapshot in Fig. 1) and written down manually. The set includes 178 domains. STARTAP was chosen since it directly provides data for both (1) and (2), and peers with the Scandinavian research network (NORDUnet)[6] ■

As discussed in Sec. 3, using the full Internet inter-domain topology $\mathcal{G}_0$ would be too demanding for our simulation platform. Therefore, the topology needs to be reduced to a smaller, manageable size. We have developed a two-step procedure to perform this topology reduction.

First, we select a domain set $\mathcal{S}$ to represent the base of our target topology.

[6]This research is conducted in Norway. The sanity tests and debugging later in our procedure are facilitated by including NORDUnet in the target topology.

This set can be selected using different selection keys, e.g. all domains in a geographic region or the Internet backbone. We have constructed the base set using the data on multicast-supportive ASes in the Internet, obtained through CAIDA (Sec. 4).

Second, we extend the base set by adding all vertices laying on the shortest path between any two vertices in the base set. We create an intermediate graph called $\mathcal{G}_1$ by including the vertices in this extended set, as well as all edges from $\mathcal{G}_0$ between all vertex-pairs in the extended set. By doing so we preserve the layout of the network, in compliance with requirements in Sec. 3. Also, the shortest path between any two vertices will be identical in $\mathcal{G}_0$ and $\mathcal{G}_1$

Now the question is how many domains should be included in the base set in order to construct the target topology of desired size. An exact key for this would be difficult to specify, since the result will always depend on the network density and the particular base set choice. We suggest the following heuristic: use 50% of the target domain number as the base set and apply the algorithm. If the resulting topology size is larger or smaller than desired, correct the base set size accordingly and repeat the algorithm.

Stage II *We know that our target network should have ∼500 domains*[7]*, while the vertex number in* $\mathcal{G}_0$ *is 10068. To construct a properly sized topology, we start with the multicast domain set* $\mathcal{S}$*, and extend it to cover domains used in (unicast) transit between domains in* $\mathcal{S}$*. We then induce a subgraph of* $\mathcal{G}_0$ *on the extended node set.*

We create a subgraph of $\mathcal{G}_0$ *that includes:*

1. *all CAIDA multicast domains from Stage I* ($\mathcal{S}$)*, 178 nodes in total*
2. *all transit domains on shortest path between all domain pairs in* $\mathcal{S}$*, adding up 163 nodes*
3. *all inter-domain links between any two ASes from the first two steps.*

This stage results in graph $\mathcal{G}_1$*, representing a network of 341 domains interconnected by 852 links.* ■

5.3 Communication Delay Model

A realistic communication delay model is a requirement for our inter-domain protocol simulations (Sec. 3). We model the communication delay through two main components: intra-domain transit delay and inter-domain communication delay. In this way we can parameterize the delay by the geographic distances (e.g. to simulate transmission time through a domain), as well as by the per-hop behavior (e.g. to model actions taken when a packet reaches a domain).

5.3.1 Intra-Domain Transit Delay

In the Internet, the AS transit delay can vary widely. We have no data describing the structure of each of the Internet domains, and, in general, no performance

[7] Our simulation problem [3] requires use of at least two routers per domain, so that the total topology would have ∼1000 routers.

measurements are published[8]. Furthermore, the transit delay will vary also depending on ingress and egress gateways for a particular data stream.

Some trivial intra-domain transit delay models can be constructed based on the topology data only (i.e. from the topology graph $\mathcal{G}_0$). For instance, assuming that larger domains will have longer intra-domain delay, we could approximate the transit delay from the domain size. The domain size could be approximated from the number of peerings for the domain, known from $\mathcal{G}_0$.

This model would however be too coarse, and could lead to significant inaccuracies. For instance, if a large domain is only "touched" by the data path (e.g. UUNET is used for transport of data stream sent from Spain to France), the same transit delay would be added as if data were sent from Spain to Australia, assuming that only UUNET is used as transport. Therefore, such a simple model is too limited for practical use.

We propose the use of geographic location of AS registration centers to approximate the intra-domain transit delay:

> Assume a data packet transits domain B on way from A to C, where A and C are domains and peering neighbors of B. Furthermore, assume that, for each domain, there is a known geographic location – the registration center – spatially within that domain.
>
> The transit delay through domain B is approximated as
>
> $$D = k \cdot d(A, C) \tag{1}$$
>
> where k is a correction constant [microseconds per kilometer] and $d(A, C)$ is a distance function that transforms the latitude and longitude data associated with each registration center to the shortest-line distance between the two registration centers.

The distance between the registration centers will often be greater than the actual link length — the actual data path is a result of careful selection by network engineers and automatic route optimizations and, in most cases, does not pass the registration center. The correction constant k must compensate this over-dimensioning. We have found that $k = 3\ \mu\text{s/km}$ is a reasonable value for achieving close-to-reality delays (Sec. 6.3).

The geographic locations of the domain registration centers can be obtained through CAIDA's NetGeo service (Sec. 4). One may wonder to which extent such locations are "central" in the geographic area covered by the domain. Furthermore, is the registration center within the physical network region at all? A European network might be registered in Bermuda. However, we have not registered any such anomalies. We have manually checked $\sim$10% of the domains in $\mathcal{G}_1$ and all were registered in a location that is believed to be within the physical area the domain covers.

This intra-domain transit delay calculation model solves the described "domain touching" problem as long as the registration centers of the ingress and the egress domains are close to the actual ingress and egress gateways. The inaccuracies may persist if more than one large domain is in the data path. For

[8] Some exceptions exist, as described in Sec. 4.

example, assume a data path passes from a European source domain A, through two large transit domains registered in the USA (B and C) to the destination domain D, also in Europe. The total intra-domain transit delay would be

$$D = d_B + d_C = d(A, C) + d(B, D)$$

where both $d(A, C)$ and $d(B, D)$ include a transatlantic distance, despite that the communication path is entirely within Europe.

To alleviate this inadequacy, we divide the large international domains from $\mathcal{G}_1$ into geographically smaller components interconnected by special intra-domain links. Due to insufficient information, we divide large domains only in up to three parts: North America, Europe and Far East. All new "subdomains" are included in our model with a unique (fictive) AS number and registration location central for the region (all European subdomains are assumed to be registered at 0° longitude, 51° latitude, North-American at (-80°, 35°) and Far East/Australian at (125°, 25°)).

Whether a domain is divided or not depends on where the domain is registered, where its peers are registered and, indirectly, on its size. For instance, a domain registered in the USA and peering only with other domains registered in the USA will not be divided regardless of its size, while a domain having both American and European peers will be put in the "international domain" set. This set is then processed on the "largest domain first"-basis, until the set is empty. (The "largest" domain is the one with the biggest number of peerings in the original topology $\mathcal{G}_0$, which is the only size-metric available.) Note that not all domains in the set are necessarily processed, a smaller domain may be excluded from the international-domain set if all its peers from other regions are processed and divided before it becomes the largest and hence the next to be divided.

The new domains and inter-domain links, together with the unchanged part of $\mathcal{G}_1$, represent our target topology $\mathcal{G}_2$.

Stage III *We modify the resulting topology $\mathcal{G}_1$ from Stage II, using the following algorithm:*

1. *Create the international-domain set I, containing AS numbers of all domains peering with at least one domain without the geographic region where the domain is registered.*

2. *While I is not empty, repeat*

   (a) *select D, the largest domain in I*

   (b) *create a new domain in each geographic region where D has peerings. At least two domains are created, since D has peerings with more than one region. At most three domains are created, because we operate with only three geographic regions*

   (c) *create mesh of links interconnecting D_1, D_2 and, if present, D_3*

   (d) *for all D's peerings, remove link $D \leftrightarrow X$ from $\mathcal{G}_1$, add $D_i \leftrightarrow X$ to $\mathcal{G}_1$ (where $i \in \{1, 2, 3\}$, depending on where X is registered)*

   (e) *remove D from $\mathcal{G}_1$*

   (f) *recalculate I, i.e., remove D and all its peers that have no other inter-domain peerings.*

In our case, I initially contained 133 domains. The set became empty after 53 iterations. The modified topology $\mathcal{G}_2$ contains 401 nodes and 915 links, and represents our final topology. ■

In the simulator, each packet entering a transit domain will be delayed by 3 μs$\cdot d(f,t)$ (according to equation (1)), where f and t are coordinates of the domains the packet originated from and is destinated to, respectively. It is therefore desirable to store the coordinates in a way that simplifies the $d(f,t)$ calculation[9].

Stage IV *In this final stage we create the geographic database to be used in our simulations. For each domain $D \in \mathcal{G}_2$, if $D \in \mathcal{G}_1$ then fetch its registration location from NetGeo database, else set the location to the default regional location (latitude, longitude):*

- *(0°, 51°), if D is European (i.e. was created by dividing a domain into two or three parts, where D is one of them and is European)*
- *location = (-80°, 35°), if D is American*
- *location = (125°, 25°), if D is in Far East or Australia.*

Store all data indexed by node number in a file that will be used in simulation run time. ■

5.3.2 Inter-Domain Delay

Two neighboring domains are in reality often interconnected using a pair of border gateways that peer through a high-performance switch. Propagation delay of such a connection is close to zero. However, the total communication delay between the border gateways can be significant in case of extensive Internet traffic load and depends also on the amount of the per-packet processing in such devices (e.g. policing, packet/flow classification, metering etc.). We have no simple means to determine the amount of this delay — a detailed simulation including background traffic would be difficult to perform both due to the lack of data describing such traffic and the increased complexity and simulation time.

We choose to set the inter-domain link delay to a fixed default value. This gives us the possibility to tune the total delay on per-hop basis, while we avoid the complexity of a realistic inter-domain delay model. We use other than default values only to account for the delay variations that can be explained from the topology and geographic data. In particular, we add extra delay to the links connecting intercontinental sites (i.e. Far East, Europe and North America).

In our model, we use empiric inter-domain delay values. We model default inter-domain delay by 2 ms, transatlantic link delay by 20 ms, and links connecting American and European domains with destinations in Far East and Australia by 80 ms delay. We stress that these values are chosen to represent typical values experienced in the Internet communications, and come in addition to the propagation delay caused by geographic distances and modeled by the intra-domain transit delay.

[9] Other methods to improve the simulation efficiency at run time are certainly possible, such as caching the previously calculated distances in a node.

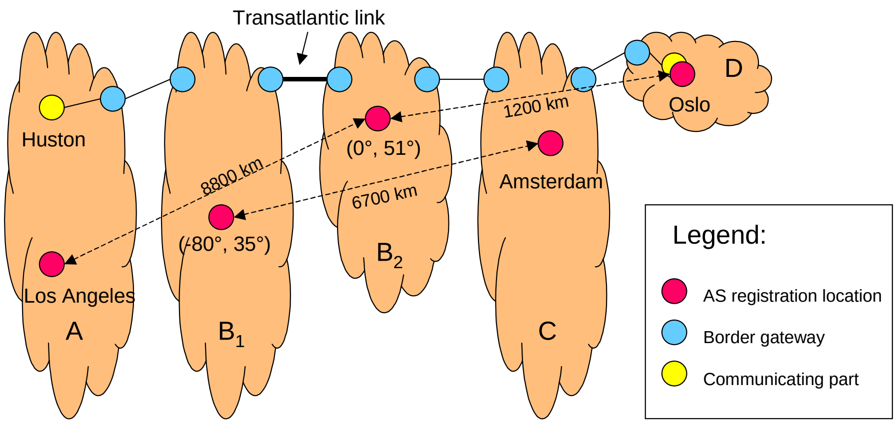

Figure 2: Delay calculation example

5.3.3 Delay Model Discussion

We have already commented the sensitivity of our delay model on geographic data accuracy. We illustrate this issue by manually calculating the communication delay between Huston, Texas and Oslo, Norway. The communicating part in Huston is connected to the Internet through a large national ISP operating its own AS ("A" in Fig. 2) registered in Los Angeles. The Oslo part is connected through a Norwegian-registered, small AS ("D" in Fig. 2).

The communication path traverses domains A, B, C and D. B is a large ISP registered in USA. It supports intercontinental connectivity and, in accordance to our delay model, has been divided in an American and a European part, B_1 and B_2. B_1 and B_2 have been assigned new registration centers, with coordinates (-80°, 35°) and (0°, 51°), respectively.

Assuming the topology layout and distances between registration centers as in Fig. 2, the simulated communication delay would be

$$
\begin{aligned}
D_{inter\_domain} &= d_{AB_1} + d_{B_1B_2} + d_{B_2C} + d_{CD} \\
&= 2ms + 20ms + 2ms + 2ms = 26ms \\
D_{intra\_domain} &= D_{B_1} + D_{B_2} + D_C = k \cdot (d(A, B_2) + d(B_1, C) + d(B_2, D)) \\
&= 3\mu s/km \cdot (8800 + 6700 + 1200)km = 50.1ms \\
D &= D_{inter\_domain} + D_{intra\_domain} = 76.1ms \qquad (2)
\end{aligned}
$$

Result (2) is close to the real latency registered on relation Oslo – Huston (~75 ms from the University of Oslo to the University of Texas at Austin). The calculation model, however, is not based on link propagation delays, queuing in routers etc., but solely on our intra-domain and inter-domain delay approximations.

In this example, the intra-domain delay in B_1 is calculated based on distance Los Angeles – London (0°, 51°), even though the data path would in practice probably never approach West Coast. The distance over-dimensioning is compensated by the low value of the correction constant k — 3 μs/km is less than propagation delay of continuous optical fiber.

The presented delay model approximates the inter-domain delay values sufficiently well for our purpose (Sec. 3). It may still be quite inaccurate in the case of large, continent-wide domains, especially if the data path includes several large domains. On the other hand, paths that include several contiguous large domains are rare both in the simulator and in the real Internet — the hop count in the shortest path routing is kept minimal, which is simple having in mind the abundance of transit network providers.

6 Validation

In this section we analyze the final topology $\mathcal{G}_2$ created using the procedure described in the previous section.

The result is in the best case as good as the initial data provided by NLANR and CAIDA. We therefore first discuss the validity of the input data we used.

Our topology was created with evaluation of multicast routing protocols in mind. We compare properties of a real, STARTAP-rooted multicast tree and the corresponding simulated multicast tree.

Finally, we compare delay data measured in the Internet with the results from our simulator as a means for delay model validation.

6.1 Input Data Soundness

The NLANR BGP routing traces represent the basis for our topology construction. All links in our target topology are extracted from this source.

A full validation of NLANR traces could not be performed, as no other similar source of information is known to us. The total number of domains corresponds to the number of domains published by Telstra [8]. We have done a partial verification by comparing the NLANR data with autonomous system registration databases (WWW.RIPE.NET and WWW.ARIN.NET), and local, familiar domains such as UNINETT (AS 224) and NORDUNET (AS 2603).

We note that the total number of route announcements by an AS exceeds the announcements available from NLANR traces by at least an order of magnitude. However, the number of announced *transport* services is close to NLANR data. For example, NORDUNET announces transport services for 14 domains, while 12 are deducible from the NLANR data[10].

In CAIDA multicast peering visualizations, the tree structure is jeopardized by additional connections between some of the nodes. For instance, in Fig. 1 at least five such connections are visible. This was somewhat surprising, as we expect to have a single path from the root to each of the leafs. The anomaly may have been caused by special routing policy (e.g. "traffic towards the root routes through neighbor D_1 if originating in this domain, and through D_2 if transit traffic"). The redundant links may appear also due to a re-routing process going on in parallel to the data collection from the involved routers, and due to routing failures. No details on graph construction method are published on CAIDA.

[10]Numbers 14 and 12 are based on data available in February 2001 and June 2001, respectively. NLANR stopped publishing BGP traces in March 2001, while RIPE announces only current AS data.

Topology name	*Number nodes*	*Number links*	*Average node degree*	*Average unicast path length*	*Network diameter*
Real ($\mathcal{G}_0$)	10329	21007	4.07	3.71	11
Final ($\mathcal{G}_2$)	401	915	4.56	3.84	11

Table 3: Comparison of the basic graph properties for the original and the target topology.

The redundant links have no influence on our topology construction procedure, as the CAIDA data is used only to create the initial set of domains to be included in the target topology.

The NLANR and CAIDA data varies slightly on a daily basis, reflecting the current situation of the inter-domain routing. These information sources cannot be fully coherent, since the data collection methods and timing are different. Our experience shows that the differences are small for data collected on the same day. Among the 178 multicast ASes that we collected from CAIDA as the base domain set $\mathcal{S}$, eleven valid ASes were not present in the NLANR trace. This can be explained by these domains not being in any path used by the 35 gateways providing the BGP trace. We have chosen to exempt these domains from our topology, and used only the data available through both sources.

6.2 Resulting Graph Properties

6.2.1 Network Density

The final topology represented by graph $\mathcal{G}_2$ was created from the global Internet topology of 10329 domains[11] and 21007 inter-domain links. Among these, 10068 "valid" nodes and 13733 "best" connections (Stage I in Sec. 5) were used. $\mathcal{G}_2$ includes 401 nodes and 915 links.

The basic graph properties for the original topology $\mathcal{G}_0$, and the target topology are shown in Tab. 3.

$\mathcal{G}_0$ and $\mathcal{G}_2$ have very similar graph properties, except the size. Our algorithm has included sufficiently many core routes to preserve the average unicast path length practically unchanged, and, due to considering the best routes only, sufficiently few routes were included to preserve the average node degree almost unchanged.

Moderate changes in network density will always be present, as the graph density varies among subgraphs extracted on different base node sets. In our case, the base set represents a state-of-the-art Internet with above-average link density. The choice of a more peripheral base node set might decrease the average node degree.

6.2.2 Multicast Tree Fanout

In our simulation project, multicast trees will be created to connect a root node with a number of receivers. We compare a real multicast tree rooted in

[11] In Tab. 1, the inter-domain topology size is 10359 domains. 30 domains were disconnected from the main topology, and ignored in our procedure.

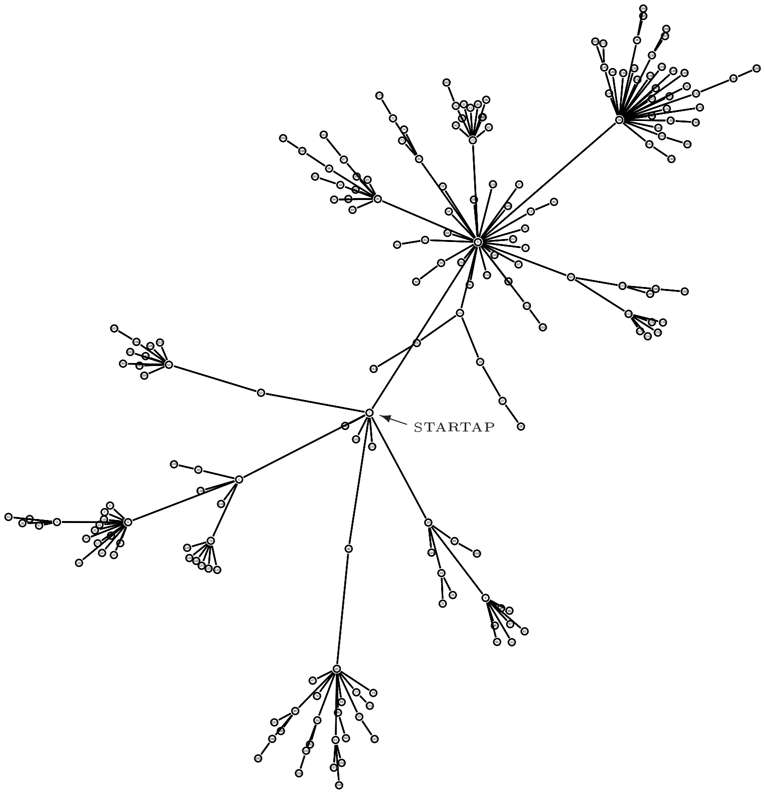

Figure 3: Tree of shortest path connections from all domains in $\mathcal{S}$ to STARTAP, simulated in our final topology $\mathcal{G}_2$.

STARTAP (Fig. 1) and a tree created by drawing shortest unicast paths[12] from STARTAP to 169 valid domains from the base set $\mathcal{S}$.

Figure 3 is a graphical representation of the resulting tree. 194 nodes are in the tree; 169 base nodes and additional 25 nodes on shortest path from STARTAP to a base node. This is close[13] to the 224-node real tree in Fig. 1. Also fanout numbers for the real and the simulated tree are similar (Tab. 4).

We stress that the two trees, although very similar in shape, do not necessarily include the same transit domains towards the leafs. This happens since shortest paths are used to connect the root and the leafs, and there are often many equal-cost paths to choose among. This has no implications on our future simulations, as the relevant metrics are unchanged and no particular routing policies will be simulated.

6.3 Unicast Hop Count and Latency

We compare the hop count and latency of the simulated and the real inter-domain topology. The comparisons are based on measurements from our de-

[12] A similar operation would be performed by a real multicast routing protocol.

[13] We have "lost" some nodes in the process of manual transcription of AS numbers, and had eleven invalid AS numbers.

Fanout	*Number Nodes*	
	Real	Simulated
0	172	140
1	18	29
2	6	8
3	10	3
4	4	2
5	2	2
6	2	1
Other	10	9

Table 4: Fanout comparison for the real and simulated tree rooted in STARTAP. Maximal fanout for the real and the simulated tree is respectively 33 and 26.

	Hop Count		*Latency [ms]*	
	Real Measurement	Simulation	Real Measurement	Simulation
Min	2.00	2.00	4.30	18.30
Mean	3.50	4.41	66.54	77.50
Max	6.00	7.00	159.50	158.10
Std	1.15	1.05	36.00	36.02

Table 5: General statistics

partment (IFI.UIO.NO, AS 224) to a set of randomly chosen destinations in our final topology $\mathcal{G}_2$. The same destination set is used both in the real measurement and the simulation.

The hop count represents the number of inter-domain links traversed on the communication path. The latency represents a one-way propagation delay from the source (AS 224) to the destination. In our *ns*-based simulator, both quantities are directly measured and logged.

The real measurements are taken using the “ping” and “traceroute” utilities. The traces are collected in ten separate measurements over a 24 hour period 6.00 AM GMT Sunday-Monday. This period of week is assumed to have the lowest network load. All latency values represent a half of the minimum RTT registered by the ping utility, in all measurements. This minimal latency value is assumed to be close to the latency in uncongested network, and, hence, comparable to the simulation results.

The real hop count value is based on the domain name changes in the path registered by the “traceroute” utility, and partially verified by manual parsing of the traces. In the traced routes, an inter-domain hop is assumed whenever the highest two levels in the domain name hierarchy change (e.g., “aa.bb.net -> cc.bb.net” is assumed to be an intra-domain hop, while “aa.bb.net -> cc.dd.net” is assumed to be an inter-domain hop).

Table 5 shows the basic statistics for the real and the simulated values. The average hop count is higher in the simulator, mainly due to the extra intercontinental hop in our model (all intercontinental domains in our model are divided in continental parts, Sec. 5.3.1). We have also registered that some new inter-domain links have appeared in the period between we had built our

Autonomous System		*Hop Count*		*Latency [ms]*	
Number	Host Name	Real	Sim.	Real	Sim.
293	chicago-nordu.es.net	2	3[§]	58	60
292	www1.es.net	3	4[§]	83	88
4	venera.isi.edu	4	4[§]	85	89
8581	ajax.noc.ntua.gr	3	7	61	152[†]
14041	ncar.ucar.edu	3	5[§]	68	89
5459	london.linx.net	3	4	21	24
3320	limes.nic.dtag.de	3	3	28	23
2833	sunic.sunet.se	2	3	4	20[‡]
7539	twnmoe10.edu.tw	5	4[§]	159	158[⋆]
5054	csoft3.prognet.com	5	5[§]	100	95
5006	postoffice.mr.net	4	5[§]	71	85

Table 6: Unicast hop count and latency comparison, real and simulated measurement. [§] All intercontinental routes include an extra hop in our topology. [†] The shortest path between UNINETT and UOI ("University of Ioannina", Hellas) passes through USA in the simulator, accounting for the high delay. The real, shortest route (European) was not present in the BGP trace. [⋆] Route used in reality is longer than the simulated due to special policing. [‡] UNINETT and SUNET speak through NORDUNET, using high-bandwidth, uncongested links in the research network. NORDUNET is registered in Denmark, adding up the delay in the simulation.

target topology and the real measurements were taken. The latency values are similar, except the minimum latency. Also the standard deviation of the results in the real and the simulated environment is similar, indicating similar result distribution in the real and the simulated system.

For a closer insight in the delay model, we have randomly selected 11 autonomous systems from our final topology, and compared the path, hop count and delay for each of the destinations in both real and simulated model. Table 6 shows that a significant difference between the real and the simulated latency was registered only when there was a mismatch between the real and the simulated AS path (AS 8581). Such anomalies are however rare in the target topology, and have a limited influence on the simulated latency value distribution.

Our comparisons show that the simulated values are almost surprisingly close to the reality. It is beyond doubt that the target topology $\mathcal{G}_2$ and the associated delay model satisfy the delay requirement from Sec. 3 and can serve its purpose, the protocol performance comparison.

7 Conclusion

In this report we have presented the method we have used to construct a topology suitable for inter-domain protocol simulations. Our final topology preserves the layout of the underlying Internet, and can be tuned to the capacity of the simulation platform.

The method is based on the reduction of the real Internet AS topology using a set of selected base nodes and a sequence of deterministic rules. The resulting topology is similar to the real inter-domain topology with respect to graph metrics such as the average node degree and the average path length. Also, the shortest path length between any two nodes in the resulting topology is equal to the shortest path length between these two nodes in the real AS topology.

In conjunction with our inter-domain topology, we have developed a communication delay model based on geographic locations of domain registration centers. This delay model yields results close to the actual delays in the Internet, and can be successfully applied in e.g. performance comparisons of inter-domain routing protocols.

This work has been motivated by a specific protocol evaluation problem and the need for an appropriate simulation model. Even though the method presented in this report is based on a number of heuristics, it yields results that are very close to the real Internet properties. We believe that the presented method deserves further investigation and improvements. For instance, determining the generality and applicability of this method to other inter-domain communication problems, as well as an exploration of the result sensitivity to various choices we have made during our work, are exciting areas for further research.

References

[1] Tony Bates, Ravi Chandra, Dave Katz, and Yakov Rekhter. Multiprotocol extensions for BGP-4. RFC 2283, February 1998.

[2] Cooperative Association for Internet Data Analysis (CAIDA). Online. HTTP://WWW.CAIDA.ORG.

[3] Tarik Čičić, Stein Gjessing, and Øivind Kure. Evaluation of global multicast protocols MSDP and BGMP. Work in progress.

[4] Deborah Estrin, David Meyer, and Dave Thaler. Border gateway multicast protocol (BGMP): Protocol specification. Internet Draft, March 2000.

[5] David Meyer and Bill Fenner. Multicast source discovery protocol (MSDP). Internet Draft, May 2001. DRAFT-IETF-MSDP-SPEC-10.TXT.

[6] National Laboratory for Applied Network Research (NLANR). Online. HTTP://MOAT.NLANR.NET/.

[7] Yakov Rekhter and Tony Li. A border gateway protocol 4 (BGP-4). RFC 1771, March 1995.

[8] Telstra. Hourly BGP routing report. Onlne. HTTP://WWW.TELSTRA.NET/OPS/BGP/.

[9] UCB/LBNL/VINT. Network simulator - *ns* (version 2). Online. HTTP://WWW.ISI.EDU/NSNAM/NS/.

[10] Ellen W. Zegura, Keneth L. Calvert, and S. Bhattacharjee. How to model an internetwork. In *Proceedings of the INFOCOM '96*, 1996.