

Modelling and Impact of BGP Routing Policies in the Internet

Maurice Rüegg

Roman Schilter

SEMESTER PROJECT

Summer Semester 2002

Tutor: Danica Vukadinović

Supervisor: Prof. Dr. Thomas Erlebach

Contents

1	Introduction	1
2	Background	2
2.1	Autonomous Systems	2
2.2	Border Gateway Protocol	2
2.2.1	Routing Tables	3
2.2.2	Path Selection	5
2.3	BGP Routing Policies	6
3	Data Acquisition	8
3.1	Data Collection	8
3.2	Representations of BGP Routing Tables	8
3.2.1	Graph Construction	9
3.2.2	Database	11
4	Path Inflation and AS Classification	13
4.1	AS Path Inflation	13
4.1.1	Multiple Vantage Points	13
4.1.2	The Route Views Project	16
4.2	Internet Structure	19
4.2.1	PQRI Classification in the Graph Model	19
4.2.2	PQRI Classification in the Path Model	19
4.2.3	Path Inflation by PQRI	21
5	Robustness	24
5.1	Fundamental Considerations	24
5.2	Algorithms	24
5.2.1	Graph Model	24
5.2.2	Path Model	25
5.3	Evaluation of Results	27
5.3.1	Development over Time	28
5.3.2	Graph and Path Model	28
5.3.3	Structure Influence	30
5.3.4	Number of Disjoint Paths and Degree	32
5.3.5	Partial Paths	33
5.3.6	Provider Core	34
6	Conclusion	35
A	Appendix	36

Abstract

The following report discusses the impact of policies in the Internet via two different models. They are called *graph model* and *path model*, are obtained both through BGP routing tables, and refer to the level of autonomous systems (ASes). While the graph model is an easy and straight-forward representation of the Internet, the path model is a more complex approach via a database storing all routing paths of an AS. Both models allow us to analyze the extent of routing policies in the Internet when classifying ASes, tracing path inflation, and testing robustness and reliability. A comparison of the two models shows strengths and weaknesses of both to be considered in future research approaches towards routing policies in the Internet.

1 Introduction

This project presents approaches to model and analyze the impact of routing policies. The studies are performed on the level of autonomous systems (ASes). An AS represents a group of hosts and routers belonging to a single administrative domain such as a company or university and has its own interdomain routing policies. These routing policies affect routing and reachability information exchanged between ASes via the border gateway protocol (BGP). Among other things, paths may be “inflated” if a route different from the shortest one is chosen because of policies.

The AS level view of the Internet has the advantage that it reduces complexity. Still, a classification of different ASes is desirable, showing their function and importance regarding the robustness of the Internet as a whole.

While previous work has concentrated on representing the Internet and its structure through a graph with ASes as nodes and links between ASes as edges, we additionally use a giant database that stores BGP paths and thus regard routing policies. We are going to show how a graph may falsify the structure of the Internet in various situations, but at the same time gives valuable information how the structure could be if no policies were applied. We call this first approach the *graph model* of the Internet.

On the other side, when disregarding resource consumption and preprocessing time of a database, the advantages of an “as is” image of the Internet and all its truly existing paths are obvious. This second concept, which we consequently call the *path model*, proves to be especially useful when classifying ASes. Of course, we compare results between the two models. They are both not perfect, have strengths and weaknesses, but combined, their view of the Internet shows the extent of BGP routing policies.

Since it is important to see through the techniques of interconnecting ASes in the Internet, section 2 provides a good understanding of the main concepts such as BGP routing and its routing tables, applied policies in the Internet, and a detailed definition of an AS.

In section 3, we show how and what information is extracted from a BGP routing table in order to get an image of the Internet. Even more important, we list the possible sources where BGP routing information is available. Then, we discuss the construction of a graph and the compilation of a database from this information.

Section 4 analyzes path inflation and reasons why the University of Oregon Route Views project serves as our main source for information. In that context, the concept of multiple vantage points as introduced in [SARK01] is examined. Also in this section is a discussion of AS classification that we adopt from [VHE01] and extend in a way to suit the path model.

Applying our database further, in section 5 we look at how robust the Internet is; we do this regarding the number of node-disjoint paths in the graph and path model. Additionally, we include the node degree into our considerations. Results are analyzed with respect to the classification found in the previous section and show that there is a handful of important providers.

Finally, we recapitulate our findings in section 6.

July 2002

2 Background

Analyzing the Internet in terms of traffic routes, topology, robustness, and relations is a difficult task considering the myriads of components that make up these terms and the countless modifications that change them every day. This chapter explains that many current studies concentrate on a somehow abstract but nevertheless accurate method to organize the Internet into autonomous systems (ASes).

Basically, this is one of two different levels one may look at to determine connectivity and routing behavior in today's Internet. On the other level, the router level, relations between individual hosts and routers are visible. Although the router level reflects the connectivity and routes between hosts much more exact, it is preferable to investigate data from the AS level in our case. There are several reasons for that:

- It is difficult to get accurate router level data of the Internet because data is subject to much more frequent changes compared to the AS level information and trusted sources are scarce.
- AS level data is more consistent because it provides an extraneous view of a group of hosts and routers and does not regard internal procedures.
- An AS is a connected group of one or more IP networks which has a *single* and *clearly defined* set of routing policies (as claimed in [HB96]). Since we are mainly interested in consequences of routing policies, it suffices to concentrate on the AS level.

2.1 Autonomous Systems

Consisting of a large collection of hosts interconnected by routers, the Internet is organized into more than 13,000 ASes. An AS is under the control of an *administrative domain* such as a company, university, or Internet service provider (ISP) and has its own interdomain routers and a clearly defined set of *routing policies*.

Each AS is represented by a 16 bit AS number, allowing for 65,535 possible ASes. The responsible institution coordinating AS number assignation is the IANA. Currently, there are more than 13,000 registered AS numbers in use. The numbers from 64,512 to 65,535 are reserved for private use and do not have to be registered.

An example of an AS is the Swiss Academic and Research Network "Switch" where all Swiss university networks are joined. Registered as AS 559, it provides connectivity to the global Internet through other ASes like GEANT, Global Crossing, or Swisscom/IP-Plus.

2.2 Border Gateway Protocol

Neighboring ASes use the Border Gateway Protocol 4 (BGP-4)¹ as defined in [RL95] to exchange routing and reachability information and provide connectivity between administrative domains, a process called *BGP route advertisement*. Each advertisement announces a route to a prefix that represents a block of IP addresses. Section 2.2.1 is going to discuss this concept in more detail.

BGP uses TCP as its transport protocol and establishes connections on port 179. There are four types of messages sent among neighboring ASes:

¹This report always refers to BGP-4 when mentioning BGP.

- OPEN opens a connection and includes information on parameters and the complete routing table of an AS.
- UPDATE transfers routing information / updates on a single route.
- NOTIFICATION informs about detected error conditions. The connection is closed immediately afterwards.
- KEEPALIVE is a repetitious message to keep connections up.

The routers in each AS apply *local routing policies* that set and manipulate the attributes associated with route advertisements. In this way, an AS administration may influence the selection of the best route for an IP prefix and decide whether to propagate routes to neighboring ASes or not. Examples of reasons for this are commercial relationships between two companies or non-profit agreements between two universities. On the other side, "disagreements" are responsible why routes may take considerable detours. [MSOP99] introduces the Routing Policy Specification Language (RPSL) that describes routing policies in detail.

2.2.1 Routing Tables

BGP advertises IP prefixes and eliminates the concept of network classes. Each prefix consists of a 32 bit address and a mask length. As an example, an institution registers 192.33.88.0/21 including 2048 IP addresses ranging from 192.33.88.0 to 192.33.95.255. Originally, 192.33.88.x to 192.33.95.x would have been 8 class C networks. Instead of 8 entries, only a single one is needed in classless inter-domain routing (CIDR), and BGP routing table size is significantly reduced.

A BGP speaking router constructs a routing table out of advertisements received from neighboring BGP-routers. Each advertisement consists of a destination network, a "next hop" IP address, and an AS path list, along with other attributes. The AS path list shows the complete route to reach a network.

A BGP router may receive differing routes (AS paths) to reach the same destination network from its neighbors. Because it is usually very well connected to others of its kind and may decide to keep older routes provided by a neighbor even when new ones arrive from it, there may be several dozen routes available for a destination network. *Import policies* are applied to filter unwanted routes and to manipulate the attributes of the remaining routes. A decision process then selects exactly one best route for each destination prefix among all the routes it received (see section 2.2.2). Finally, *export policies* manipulate attributes and decide whether to advertise a route to a neighboring AS.

Hence, the routing table of a BGP-speaking router contains reachability information on a very large portion of the Internet in its AS path list to all destination networks. It does not contain the complete topology of the Internet, but rather a "centralistic view", itself being the starting point for all traffic.

BGP routing tables include millions of entries and may become several hundred megabytes large. Figure 1 shows the beginning of such a BGP routing table taken from the AS of the University of Oregon on May 1, 2002. There, two different paths to the destination network 3.0.0.0 are shown (lines 6 and 7). The second one is marked as the best route by *> which leads through AS 7018 to the destination AS 80. To reach the next hop, AS 7018, IP address 192.205.31.33 has is used.

With the BGP routing table in figure 1, we explain some important points:

```

1 BGP table version is 276469, local router ID is 198.32.162.100
2 Status codes: s suppressed, d damped, h history, * valid, > best, i internal
3 Origin codes: i - IGP, e - EGP, ? - incomplete
4
5   Network          Next Hop          Metric LocPrf Weight Path
6 * 3.0.0.0          213.200.87.254    930           0 3257 7018 7018 80 i
7 *>                192.205.31.33     0             0 7018 80 i
8 [...]
9 * 9.141.128.0/24  209.244.2.115     0             0 3356 9057 2686 ?
10 *>                213.200.87.254    10            0 3257 2686 ?
11 *                217.75.96.60      0             0 16150 8434 8434 2686 i
12 [...]
13 * 129.132.0.0    209.244.2.115     0             0 3356 9057 3303 559 i
14 *                216.140.8.63      0             0 6395 3303 3303 559 i
15 *                213.200.87.254    90            0 3257 559 i
16 *                203.62.252.26     0             0 1221 4637 3303 559 i
17 *                195.66.224.82     0             0 4513 3303 559 i
18 *>                212.4.193.253     0             0 8918 559 i

```

Figure 1: Example of the beginning of a BGP routing table, taken from the AS of the University of Oregon on May 1, 2002.

- Line 2 shows possible status codes. The most important code is the combination `*>` which marks the valid best route for a given prefix.
- Every route ends with an origin code described in line 3. IGP stands for Interior Gateway Protocol, EGP for Exterior Gateway Protocol, incomplete means routes whose origin is not known.
- *Metric*, *LocPrf*, *Weight* are (partially proprietary Cisco-specific) parameters used to determine the best route (see section 2.2.2).
- *Path* lists the ASes a route touches to reach a network. There are two special cases that may appear:
 - *AS prepending* is a technique to elongate a route and make it less attractive to others. In line 11, AS 8434 appended its AS number twice before forwarding routing information to AS 16150 in order to discourage AS 16150 from using the path via AS 8434. This is perfectly legal.
 - *AS Loops*, on the other hand, are illegal. Our example table is loop-free. However, loops appear once in a while and if line 14 showed a path of **6395** 3303 **6395** 559, we would have found one.

A last, very important aspect when analyzing BGP tables is *route aggregation* because it may perturb paths and their length. It is a technique illustrated in figure 2 where a larger *provider AS* decides to hide some of its *customer ASes* to the outside. Instead, the provider acts as if all its customers' IP networks were its own. The AS path in a BGP routing table thus ends at the provider, even though there would be one more hop in the path to the aggregated customer. A detailed explanation of route aggregation is provided in [Cisa]. Section 4.1.2 is going to show further effects of route aggregation.

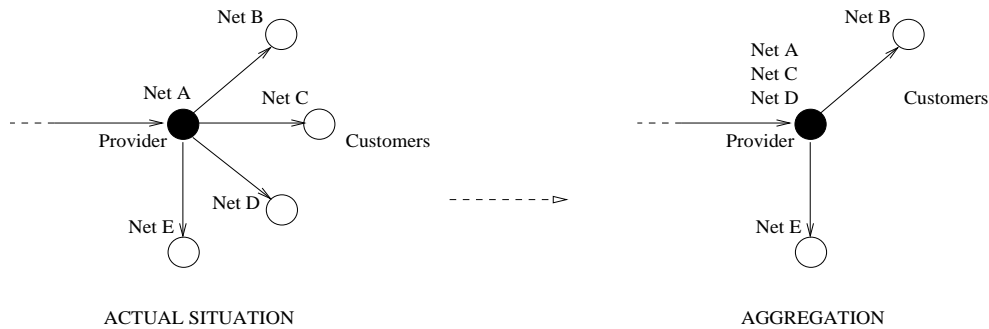


Figure 2: AS route aggregation.

2.2.2 Path Selection

[RL95] does not define an exact decision process to select a best route for a destination prefix if several routes are available. It only specifies a "tie-breaking procedure". However, router vendors adhere to a de facto standard to facilitate interoperability between different products ([Cisb], [Jun]). For each prefix in the routing table, the BGP routing protocol process selects a single best path, called the active path. It is determined through the following priority list:

1. **Highest local preference** — Prefer a route with the highest local preference, where local preference is assigned by the import policy.
2. **Shortest AS path** — Prefer a route with the shortest AS path length, as conveyed in the BGP advertisement.
3. **Lowest origin code** — Prefer a route with the lower origin code. Routes learned from an Interior Gateway Protocol (IGP) have a lower origin code than those learned from an Exterior Gateway Protocol (EGP), and both these have lower origin codes than incomplete routes.
4. **Lowest MED** — For routes with the same next-hop AS, prefer a route with the smallest multiple exit discriminator (MED) value, as conveyed in the BGP advertisement or reset by the import policy.
5. **eBGP over iBGP** — Prefer strictly external (eBGP) paths over external paths learned through interior sessions (iBGP) since leaving the AS directly is preferable to forwarding traffic through the AS to another router.
6. **Lowest IGP metric** — Prefer a route with the smallest interdomain metric to reach the next hop since this enables each router to select its "closest" exit point.
7. **Oldest route** — Prefer the route that was received earliest since this route is more likely to be stable.
8. **Lowest router id** — Prefer the path that was learned from the neighbor with the lowest peer IP address.

2.3 BGP Routing Policies

Local BGP routing policies have been mentioned in section 2.2. Moreover, local export policies have a direct impact on the AS paths seen from a particular AS and at the same time define global relationships between ASes. One speaks of either *provider-customer* or *peer-peer* relationships depending on contracts that define exchange of traffic and pricing models between domains. Each AS sets up its export policies according to its relationships with neighboring ASes. This results in three cases that govern export policies:

- **Exporting to a provider** — To a provider, an AS may export its own and its customer routes, but *not* its provider or peer routes.
- **Exporting to a customer** — To a customer, an AS may export all its routes.
- **Exporting to a peer** — To a peer, an AS may export its own and its customer routes, but *usually not* its provider or peer routes.

The graph in figure 3 illustrates this process. Because of the form of this graph, AS relationships are often referred to as *valley-free* consisting of an *uphill portion* and a *downhill portion* and at most one peer-peer link. This policy is also called the *selective export rule* [Gao01], [SARK01].

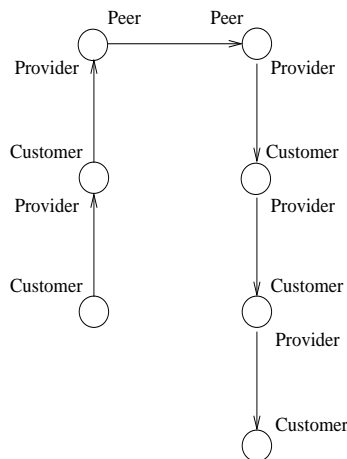


Figure 3: Example of the valley-free export policy rule.

A second situation that reflects the use of policies is illustrated in figure 4. It is called the *prefer-customer rule* since an AS typically prefers a customer route to a route via a provider or peer. This is favorable because a provider AS does not have to pay its customers to carry traffic and at the same time may avoid traffic congestion at peering exchange points.

To conclude, the policies mentioned have a great impact on AS routing in the Internet. We are going to see this clearly when discussing path inflation and AS classification in section 4 and when looking at node-disjoint paths between ASes in section 5.

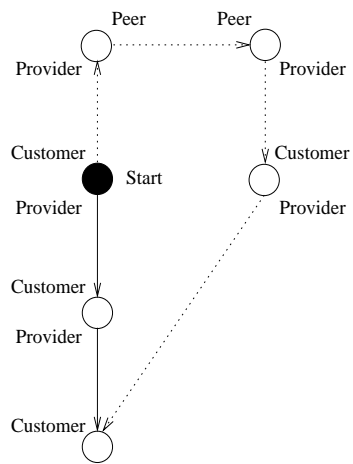


Figure 4: Example of the prefer-customer policy rule.

3 Data Acquisition

As described in the introduction, we aim to investigate the impact of routing policies in the Internet in terms of path inflation, AS classification, and robustness. To achieve this, we look for simple representations of the Internet which only include information that is relevant for our analysis. In the following, the way to acquire this information is described and what properties are extracted from it in order to meet our requirements.

3.1 Data Collection

There are several institutions in the Internet which operate a so called looking glass server. These servers allow us to obtain their view of the Internet by downloading their BGP routing tables at every moment. One of the biggest and best service of this kind is the Route Views Project from the University of Oregon [Mey]. It provides current BGP routing tables every two hours and offers historical data, too. Due to its remarkable grade of connectivity with other ASes (presently 51 direct neighbors), it offers an extensive view of the Internet's topological structure and the existing routings.

However, we have seen in section 2.2.1 that an AS may not have a BGP routing table covering the entire Internet topology and therefore may lack some ASes and routes it is not able to “see”. Hence, it may be favorable to take BGP routing tables of several servers into consideration to receive a more complete image of the current Internet topology. In the literature, this proceeding is known as looking from *multiple vantage points* [SARK01]. In section 4.1.1 we are going to dwell on the question if the use of multiple BGP routing tables is helpful in our case.

Another question to be asked is how much data shall be made use of to get a good base for our analysis. The contents of a BGP routing table is changing persistently due to the announcement of new, the withdrawal — permanently or only temporary — of invalid and the modification of existing routes. In other words, a BGP routing table is only a snapshot and never includes all available connections and routes. It may thus be useful to look at more than one table to see the changes and developments over time. Tests, where we downloaded BGP routing tables every six hours and looked for changes, have shown that in our case, it is of no use to compare tables in too short intervals. In this context, an interesting analysis of BGP routing table dynamics discussing aspects like table growth, update frequency, and prefix length distribution can be found at [Hus].

For our work, we mainly use data from Route Views. We downloaded the BGP routing tables of the first day of every month from May 2001 until May 2002 to investigate development over time. In addition, we use data from 13 different ASes dated April 6, 2002 from [Aga] to analyze the concept of multiple vantage points.

3.2 Representations of BGP Routing Tables

A raw BGP routing table whose look has been presented in section 2 is not suited for an efficient and manageable analysis. The same path may occur several times in a table and the size of Oregon's table for example is about 450 megabytes. We are forced to extract the relevant information from the tables and store it in an appropriate form. We are going to do that in two different ways:

- Representation as a graph: Every AS is a node in the graph and two consecutive ASes in a path generate an edge between the two ASes. This is the usual way to represent the Internet Topology.
- Storage in a database: Each path is stored in a database along with its IP prefix and a tag marking whether it is a best route or not.

In the following, we describe the procedure to get the two forms of representation and show their advantages and disadvantages in view of our intention to investigate routing policies. For both representations, private AS numbers (64,512 to 65,535) and AS prepending are ignored.

3.2.1 Graph Construction

The usual way to represent the Internet's topology is to create a graph. For simple handling of graphs we use the LEDA library for C++ [Alg]. It offers a data type `graph` and valuable graph algorithms. Our proceeding to create a bidirected graph from a BGP routing table is as follows:

- Each AS number is extracted from the path list and one single node is created for each AS.
- Two consecutive ASes in a path list result in two edges of weight 1 between the respective nodes, one in each direction. Hence, it is assumed that two connected ASes may always communicate in both directions, even though the other direction does not show up in the path list and might not be used.
- The number of the AS the BGP routing table has been downloaded from is prepended to each path, since the paths start at this AS.
- All paths — not only best routes — are used to create the graph since we want to have an image of the topology including all known nodes and edges.
- Multiple parallel edges are removed, i.e. there is only one edge between two nodes with the same direction.

Figure 5 shows as an example a digest of a BGP routing table from Oregon and figure 6 the resulting graph of the construction procedure. Node 3582 represents the AS of the University of Oregon and marks the start of each path. All edges appear in both directions and the loop caused by the path on line number 2 in figure 5 from AS 17691 to itself is omitted since AS prepending is ignored.

```

1 * 218.40.16.0/20 157.22.9.7 0 715 7091 4725 17691 i
2 * 202.232.1.91 0 2497 17691 17691 i
3 * 192.205.31.33 0 7018 701 4725 17691 i
4 *> 216.140.14.186 0 6395 4725 17691 i

```

Figure 5: Digest of a BGP routing table (example 1).

This operation is carried out separately for each BGP routing table and results in 26 graphs (May 2001 until May 2002 from Oregon and 13 tables from April 6, 2002) with up to 13,000 nodes and 55,200 edges (i.e. 27,600 pairs of connected nodes).

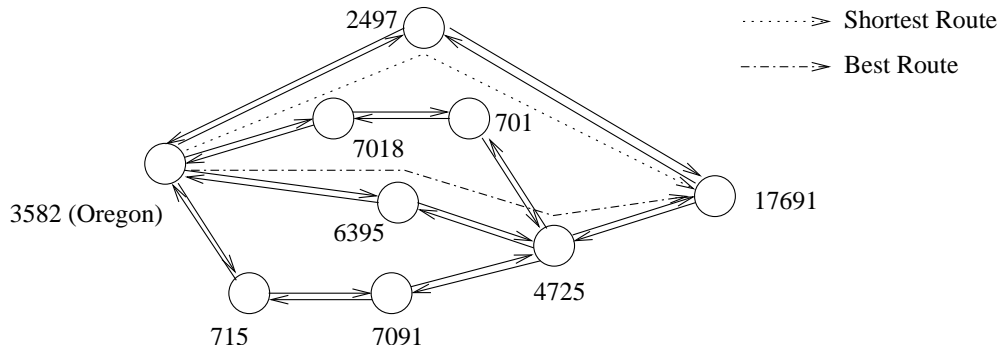


Figure 6: Graph created from BGP routing table in figure 5

The advantage of a graph is that we may apply algorithms to find shortest path or maximum flow in an easy and efficient way. Yet, graphs have a serious drawback when keeping in mind that we aim to investigate routing policies and their consequences. By creating a graph, valuable information about which paths are preferred (best routes) is getting lost. This phenomenon may be illustrated by the example in figure 5 and 6. The best route marked by `*>` on line number 4 needs 3 hops to reach destination AS 17691. However, there is a shorter path on line 2 which takes only 2 hops. Thus, when calculating the shortest path between Oregon and AS 17691 in the graph we get a value of 2 hops although the preferred best route according to policies requires 3 hops.

Another phenomenon we have to pay attention to is the formation of new paths, i.e. paths which appear nowhere in the BGP routing table. Figure 7 and 8 clarify this problem. The best path on line 3 needs 4 hops to destination AS 7499. However, when we look for a path to the second last AS 9269, we encounter a path with only 2 hops (line 6). This means that AS 7499 may be reached in 3 hops when ignoring policies. The graph in figure 8 illustrates this insight.

```

1 * 202.182.252.0/22 209.244.2.115 0 3356 9225 9269 9269 7499 i
2 * 216.140.8.63 0 6395 3356 9225 9269 9269 7499 i
3 *> 157.130.185.17 0 701 703 9269 7499 i
4 [...]
5 * 203.80.64.0 209.244.2.115 0 3356 2828 9269 i
6 *> 205.158.2.126 0 2828 9269 i

```

Figure 7: Digest of a BGP routing table (example 2).

Another cause that falsifies information about routes and reachability in our graph is the fact that we generate two edges in both directions between two nodes even if there exist only paths using a link in one single direction. One might object that we should create only edges in directions that appear in the path list. Only about 740 of the 27,600 connected pairs mentioned above really have corresponding links in both directions in the path list. Nevertheless, we continue to create links in both directions. The following successive considerations lead to this decision:

1. As we have seen above, new routes — in terms of policies possibly invalid — between

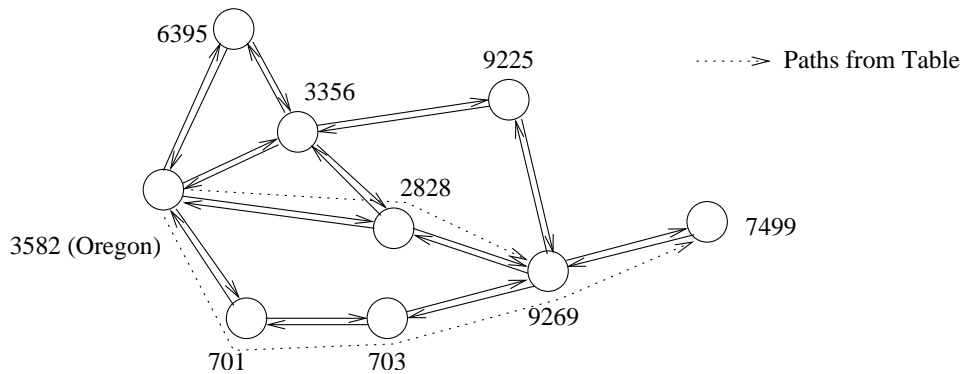


Figure 8: Graph created from BGP routing table in figure 7

two stations arise in any case even if only links are created in the graph which really exist in the path list.

2. Thus, we make use of the graph representation as a *model of the Internet's topology without attention to policies*. The graph shall include as many connections as possible that physically exist.
3. Hence, the graph may provide us with information about how *circumstances would be if there were no restrictions due to policies*. For example, we may calculate the *really* shortest path between two nodes by using Dijkstra's shortest path algorithm.

In the following, we are always going to speak about the *graph model* when using the graph for our analysis. Since the graph may not make allowance for the limitations due to routing policies we need a representation that measures up to them. From this, it follows that we have to *retain the structure of the paths* and store them in an appropriate form.

3.2.2 Database

With the aid of a database we are able to store our data and — being the crucial point for our application and the real advantage compared to the raw BGP routing table — to query them in an easy and flexible way. We use the popular open source database MySQL [MyS] because of its flexibility and wide field of possible applications.

For our analysis, we store each path along with the IP prefix and a tag marking whether it is a best route or not. Additionally, a timestamp indicates the date of the BGP routing table the paths have been extracted from. The storage of the paths entails some questions because of their variable length. We want to be able to easily access an AS number at an arbitrary position within the path. Especially, it must be possible to offhand find the destination AS. Furthermore, we are forced to keep memory requirements as low as possible since one single BGP routing table contains about 5.5 million paths and a path may include more than 8 AS hops.

As a result of these demands, we store each path in a single binary field. Each AS number is represented by 2 bytes, enabling $2^{16} = 65536$ possible values what is exactly the range of valid AS numbers. By means of this storage method, we may for example access the

destination AS by query `right(path,2)`, i.e. the two rightmost bytes. Analogically to the graph model, we are going to refer to the *path model* when employing our database.

It is not only the storage of paths that makes our database useful, but also the ability to store results of our calculations and analysis. We create another table containing the following values:

- longest, shortest, and average path length to each destination AS, extracted from paths
- shortest path to each AS, calculated by Dijkstra's algorithm in the graph
- structural type of AS, according to PQRI classification, on the basis of graph and paths, respectively (see section 4.2)
- number of edge- and node-disjoint paths to each destination, extracted from graph and paths, respectively (see section 5.2)

The MySQL files containing this data set are available for public download at <http://people.ee.ethz.ch/~mrueegg/BGP/>.

4 Path Inflation and AS Classification

The following experiments are based upon results in [GW01] discussing AS path inflation through routing policies. They also take into consideration observations on the advantage of multiple vantage points, a concept introduced in [SARK01]. To further investigate AS path inflation, the simple but effective division of ASes into P, Q, R, and I nodes, described in [VHE01], is extended and applied.

4.1 AS Path Inflation

Previous studies in [GW01] have defined AS path inflation as the difference between the chosen AS path (best route) length and the Dijkstra shortest path for a pair of ASes. Of course, to be able to find a Dijkstra path between two ASes, a graph needs to be created as described in section 3.2.1 where we introduced the graph model. Since an AS pair might communicate over different AS paths for different destination network prefixes, the longest of these paths — we call it *longest best route* — is taken as the one to compare to the Dijkstra path; this points up the extent of AS path inflation at its extreme.

Additionally, we are interested in the *shortest best route* which shows the best case that policies allow for. In the following, we present various comparison models for AS path inflation.

As an example of path inflation, we use the routing table in figure 7 from the previous section. There, figure 8 shows a graph constructed from that table. In figure 9, it may be seen that the best route from AS 3582 to AS 7499 includes one hop more than the path found by Dijkstra’s algorithm in the graph. We have found a path with an inflation of one hop.

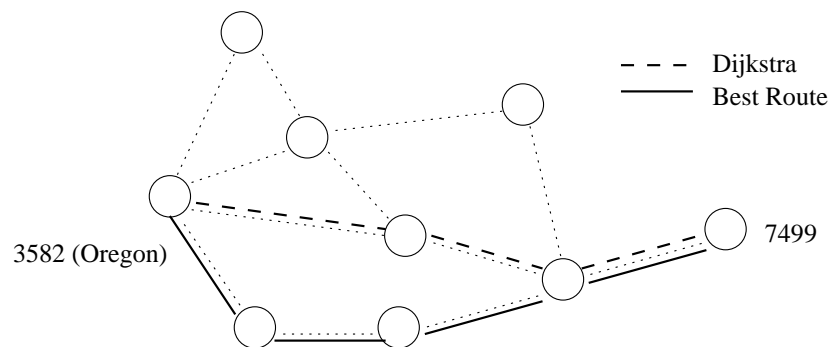


Figure 9: How Dijkstra finds a better path than the best route of a routing table.

4.1.1 Multiple Vantage Points

[SARK01] introduced the concept of multiple vantage points to get different views of the Internet at the same time. We are going to use their most recent data set from April 6, 2002 to examine how different vantage points affect AS path inflation and from where routing policies may be seen the clearest. This data includes BGP tables of 13 looking glass servers listed in table 1 including the one from the University of Oregon which is going to play an important role afterwards.

AS #	Name	Visible ASs	Links
3582	University of Oregon	13049	54826
1838	CERF Net	12780	35864
3549	Global Crossing	12779	32854
3967	NetUsaCom / Exodus	12839	43982
4197	Global OnLine Japan	12793	32140
5388	Energis Squared	12794	34078
5511	France Telecom	12791	34324
6539	GT Group Telecom Services Corp	12743	32962
7018	AT&T Internet	12739	34298
8220	Colt Internet	12799	37342
8709	RIPE / Exodus	12806	37250
9328	Asia Pacific Network Information Centre	12817	38496
15290	AT&T Canada	12792	33906

Table 1: The University of Oregon Route Views project and 12 other looking glass servers on April 6, 2002.

It is obvious that examining path inflation for the various looking glass servers in table 1 will return results that are very different from each other because those vantage points have significant differences in visible ASes and known links. Still, it is very interesting to look at their different ratios of path inflation.

In order to get a diverse and multi-angled picture of the chosen vantage points, the path inflation definition from above is varied in the following. First, we do not only look at the longest best route to a certain AS, but also at the shortest. Results of this are shown in figure 10. Looking at the overall percentage of inflated paths, considerable differences are observable. While AS 5388 has 49.7% and AS 9328 47.7% inflated longest best routes, AS 5511 has only 14.4%. Also, whereas Oregon has almost three times as many longest best routes inflated as shortest best routes, AS 5388 has only a fifth more.

Figure 11 shows that an inflated path includes between 1.17 and 1.28 additional hops on average which implies that most inflated paths are inflated by only one hop.

Another surprising fact in figure 10 is that path inflation of the shortest best route compared to a Dijkstra path is at least 4.89% for AS 5511 and at most 40.0% for AS 5388. This means that for AS 5388, almost half of its shortest best routes could be shorter. To analyze this in more detail, a comparison between the shortest best route and the truly shortest route not necessarily marked as best is made. Again, the base measurement for path inflation is a Dijkstra shortest path from our graph model. Figure 12 shows that even in that case, there is a significant difference of up to 3.7% between paths due to routing policies.

One may notice that AS 9328 shows some strange behavior. Results show that there are more truly shortest paths inflated than shortest best routes. This is explained by the fact that the routing table of AS 9328 did not indicate a best route for some IP network prefixes it listed at the time of this analysis.

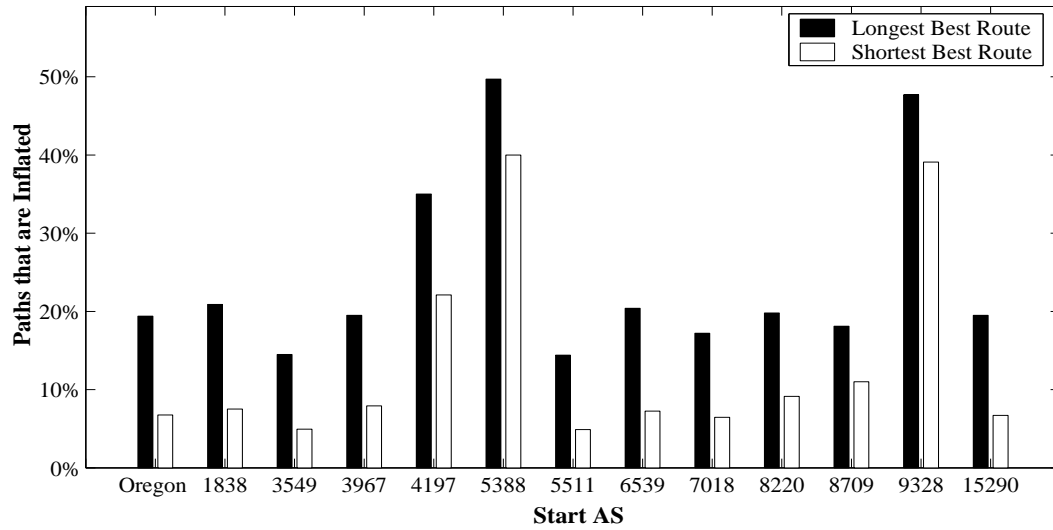


Figure 10: Path inflation for different vantage points.

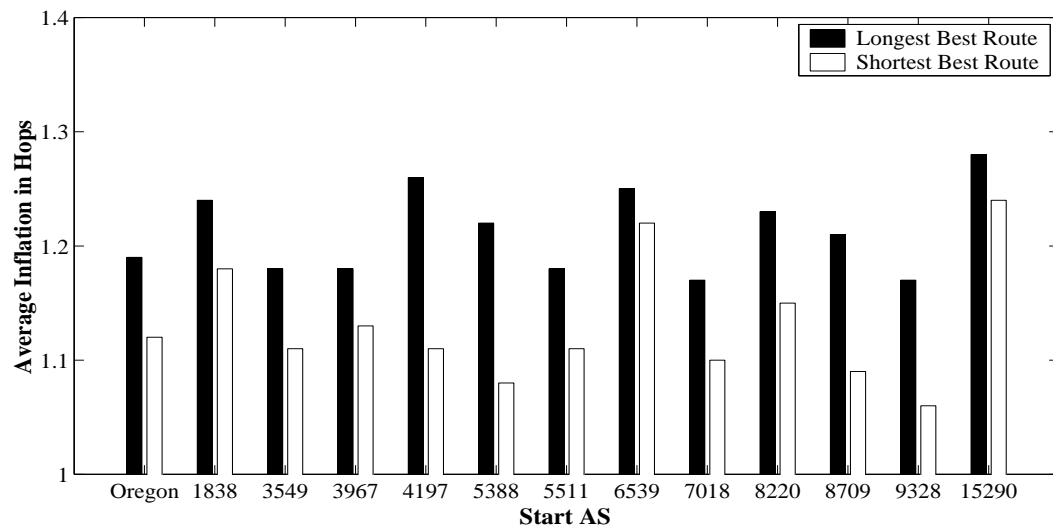


Figure 11: Average inflation in hops for inflated paths for multiple vantage points.

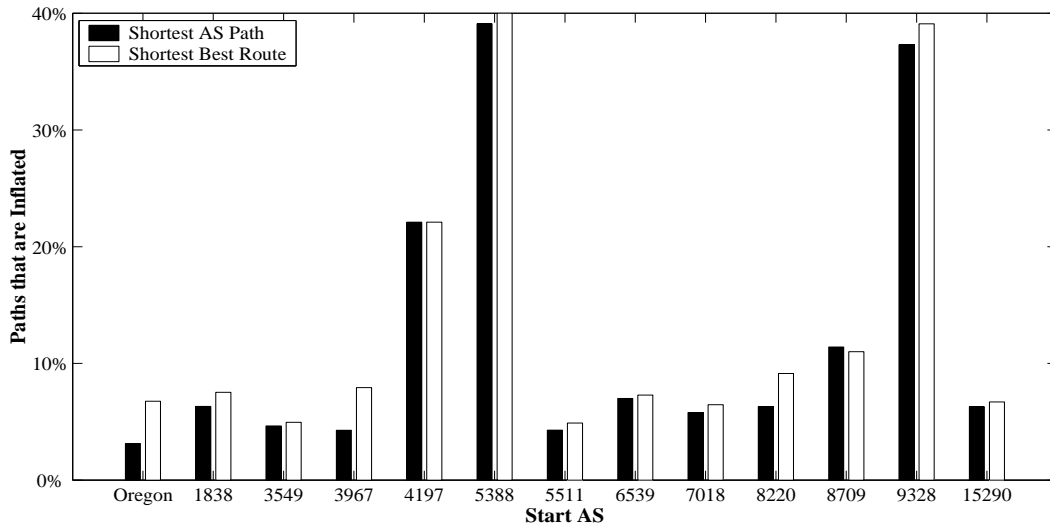


Figure 12: Path inflation comparison between shortest best route and the truly shortest route to a destination in a routing table.

4.1.2 The Route Views Project

We have shown that there are considerable differences between looking glass servers regarding how they see the Internet. In order to get the best view, two approaches are possible:

- Join the data from multiple vantage points.
- Choose the one vantage point with the most complete view.

Because of the sheer size of a BGP routing table, the first option seems not very attractive. This is the reason why the University of Oregon Route Views project [Mey] was called into life (see section 3.1). To prove that this vantage point has the best view of the Internet, we *joined the data* of all 13 looking glass servers from table 1. In other words, while a certain AS is still the “start point” of our observations, it now has additional information about Internet topology from twelve other ASes leading to possibly additional nodes and especially edges in a graph. Calculating path inflation for this new view, we compared results to those of single vantage points. Figure 13 shows that for some ASes like 4197, longest best route inflation is affected by up to 18.1%.

However, for AS 3582 (Oregon) the additional information from the other ASes has almost no effect. Since only 0.1% more inflated paths may be found, it implies that Oregon has a very good view of the Internet by itself and that it is well suited as starting point when observing BGP routing policies. Because of this and the easily available data, we are going to concentrate on this AS exclusively in the following.

Let us look at path inflation due to routing policies for the Route Views project. Figure 14 shows how path inflation decreased slightly over the last year. This is true when looking at the longest best route to an AS as well as for the shortest best route. The difference between shortest and longest best routes stays almost constant at around 12%. This implies that there is still a heavy imbalance of path length to different networks on the same AS.

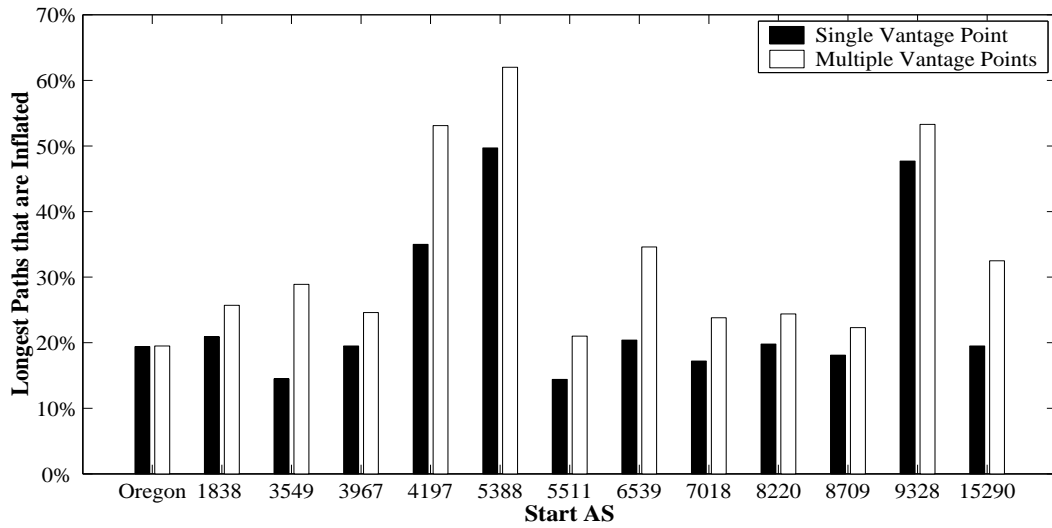


Figure 13: Path inflation comparison for single and multiple vantage points.

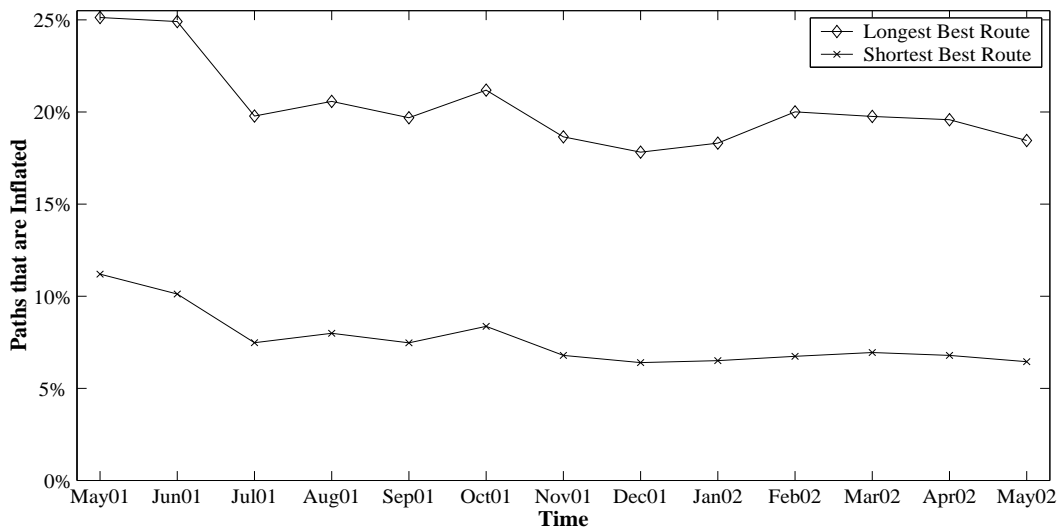


Figure 14: Path inflation data of AS 3582 (Oregon) over a year.

What is more remarkable is the break after May and June 2001 where path inflation dropped by 5%. This phenomenon is also visible in figure 15 where we count the number of ASes which have the same number of hops in their longest best route. The data shows the same tendency as the graphs in figure 14. While in May 2001, there were considerable more ASes with an absolutely longest best route of 3 hops than six months later, the data does not change much for the following six months.

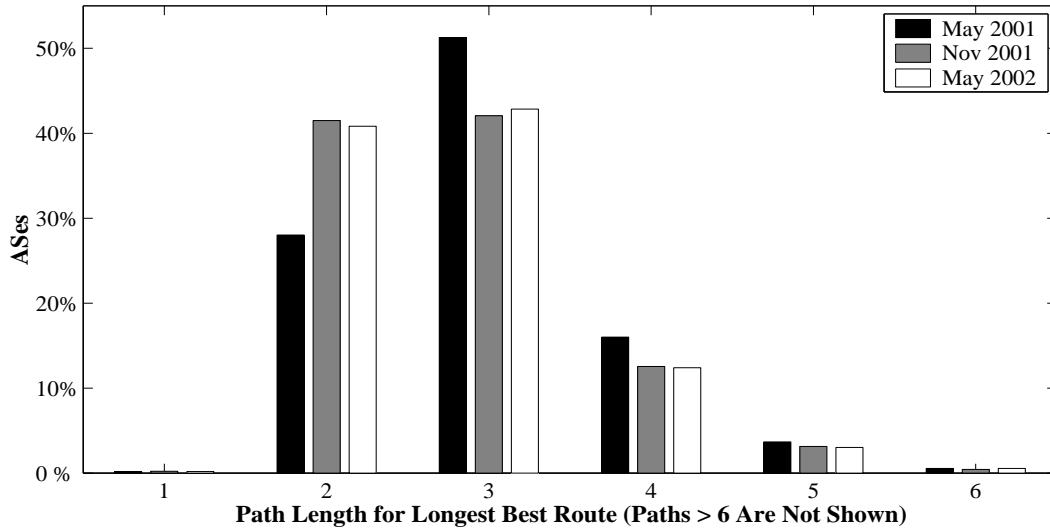


Figure 15: Path length distribution — development over time.

The results in figure 14 and 15 made us look for reasons of the mentioned trends. We came up with the following possible causes:

- *Route aggregation* has gained significant importance for BGP routing over the last year. This technique allows a provider to incorporate its customers by not advertising them as stand-alone ASes, but acting as destination AS for their network prefixes. Thus, certain destination ASes may exist and be registered at IANA, but hidden from looking glass servers. Obviously, they affect the image of the Internet we obtained from the Route Views project.
- The Route Views project is constantly expanding the list of peers it is working and exchanging data with. Thus, data may not always be *homogenous* when looking at larger time slots.
- There exist a trend to merge existing ASes because of better manageability and economic reasons and to integrate new networks into existing ASes instead of creating new ones. This development may reduce the number of “oddly” configured BGP routers and policies.

Generally, we are looking at a small time window and do not know whether the observed development will continue and how much of our results is due to statistical deviation.

Taking our consequences from the above findings, we decided not to show any developments over time for further evaluation of routing policies in the following section 4.2.3, especially since results of those evaluations mirror the results from 14 and 15 exactly.

As a general statement, we may say that path inflation depends very much on the data set one uses and on what information one extracts from these data sets. Different ASes maintain different BGP routing tables where path inflation varies from 15 to 50% when extracting the longest best route and from 5 to 40% for the shortest best route. Whether extracting the longest or the shortest or any other route returns meaningful results remains an open question. Additionally, a comparison of AS hops is sometimes not meaningful because data may travel over more internal router hops in a large AS than in a small one before leaving for the next AS. There, looking at the router-level may provide more accurate results.

4.2 Internet Structure

Looking at the Internet topology from an AS point of view reduces complexity significantly. Still, there are more than 13,000 ASes and it is clear that not all of them are of the same interconnectivity, the same importance, the same reachability, or the same hierarchy. Early models to solve this predicament have classified ASes based on degree. ASes with a large number of neighbors (a high node degree) are placed above ASes with fewer neighbors. Today, this would result in a somewhat distorted image of the Internet where many nodes are multi-homed but small ASes, i.e. customers of more than one provider. They would be given too much importance in such a degree-based model.

More recent approaches to structure the Internet include more complex methods, all trying to capture interconnection and relationship between ASes more accurately. We decided that it is important to take into account this structure when looking at routing policies and chose to use one such classification method, called the PQRI model introduced in [VHE01].

4.2.1 PQRI Classification in the Graph Model

In [VHE01], the classes P, Q, R, and I are defined in graph-theoretical terms:

- **P** contains all nodes with degree 1 (*leaves*).
- **Q** contains all nodes that are neighbors of P-nodes.
- **I** contains all nodes which are isolated after removing all P- and Q-nodes.
- **R** contains all non-isolated nodes after removing all P- and Q-nodes.

Practically speaking, P-nodes are single-homed stubs or customers while I-nodes are multi-homed stubs or customers. Some Q-nodes are providers and the best connected part of the Internet because it is in their interest to deliver interruption-free and direct connectivity to the Internet for their customers. The largest component of R-nodes builds the Internet core together with the largest Q-nodes while the others components of R-nodes may be seen as AS alliances. A fictive example of a graph with classified nodes is depicted in figure 16.

4.2.2 PQRI Classification in the Path Model

There are some modifications we suggest when translating PQRI Classification from a graph model to a path model of the internet. While the definitions for P- and Q-nodes hold perfectly, I-nodes in a graph may not always be multi-homed stubs, but sometimes forward traffic (see figure 17). In our analysis, we found that this is true for 1% of I-nodes.

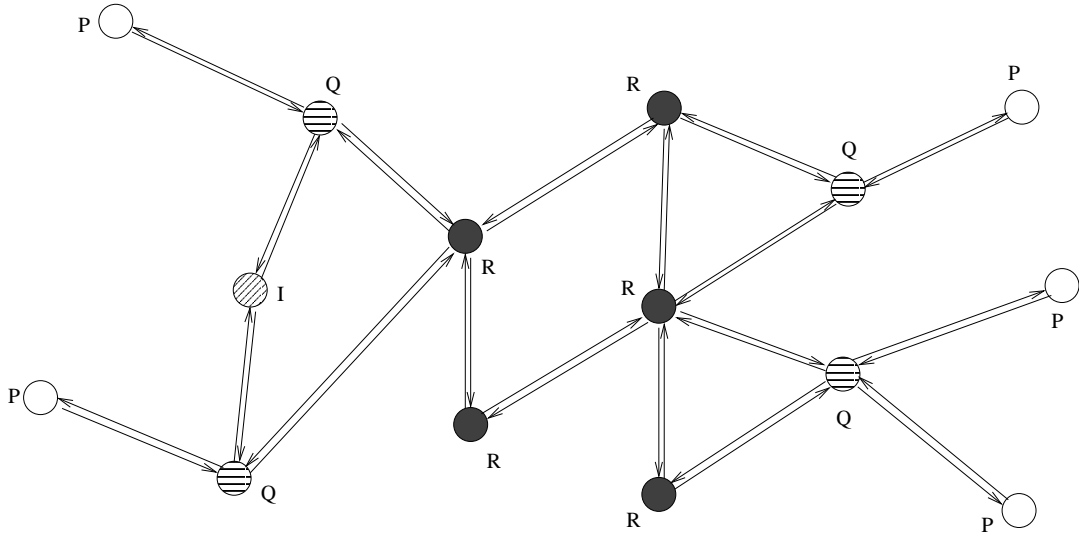


Figure 16: Example of how nodes of a graph might be connected and the classes they belong to.

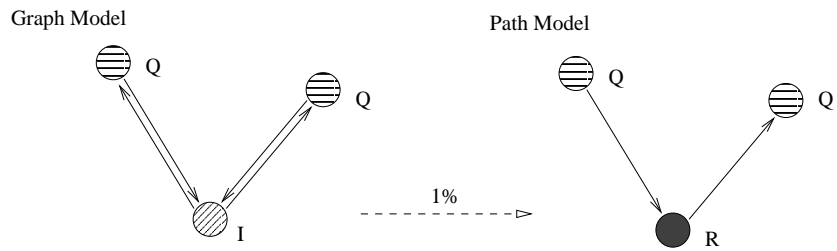


Figure 17: I-nodes may become R-nodes in the path model.

Similarly, there are two cases, where R-nodes have to be counted as multi-homed stubs in a path model. I-nodes still contain all ASes which are isolated after removing P and Q ASes and do not forward traffic. Additionally, they include ASes that are connected to R- and Q-nodes, but only as destinations, which is true for 55% of R-nodes. Analogically, there are 5% of R-nodes we now count as I-nodes which are solely connected to R-nodes, again only as destinations. The two cases are shown in figure 18.

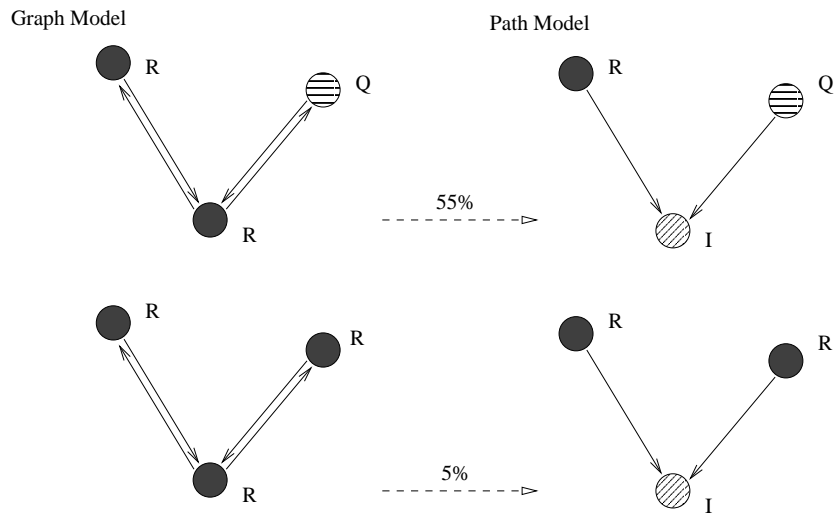


Figure 18: R-nodes may become I-nodes in the path model.

The nodes of our example graph from figure 16 undergo some changes when applying these new rules of classification for a path model because of policies. The resulting classification network is shown in figure 19.

An illustrative example of an AS that is located in the core network and identified as an R-node in the graph model is AS 12076. It has 27 direct connections to other R- and Q-nodes and is reclassified as a multi-homed stub in the path model, that does not forward any traffic. AS 12076 hosts Microsoft's MSN service.

[VHE01] says that more and more ASes in the Internet are multi-homed stubs (I-nodes) because of an increasing demand for multi-homed access to ISPs. As a direct conclusion to the above results, we can confirm this proposition and even emphasize it. Figure 20 shows that for our path model, multi-homed stubs are even more widespread than assumed from results in the graph model.

4.2.3 Path Inflation by PQRI

Results from figure 20 lead us to the question how path inflation depends on the node type of the destination AS. Figure 21 shows path inflation considering the longest best route. A comparison is made between AS classification in the graph model and in the path model.

The most interesting points in figure 21 are a clear indication that R- and Q-destinations have more inflated paths than Is and Ps. This indicates that providers (Q-nodes) and core ASes (R- and Q-nodes) have more differentiated and complicated routing policies than customers (I- and P-nodes). They have more connections among each other and are better

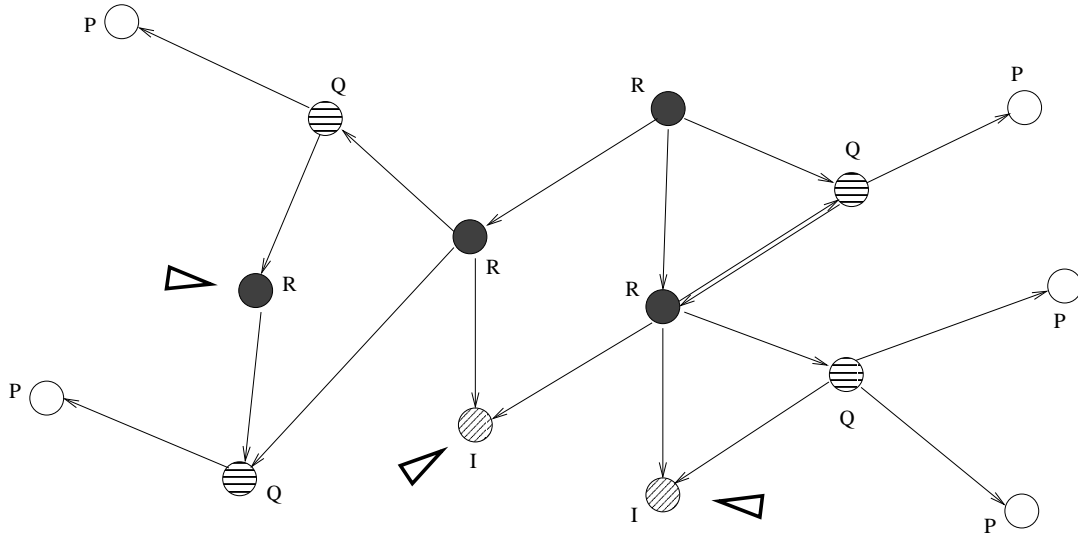


Figure 19: Example of how ASes might be connected — considering routing policies — and the classes they belong to.

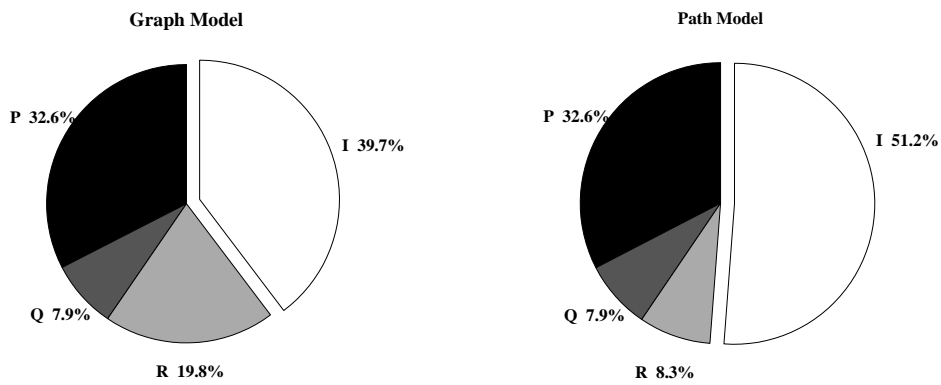


Figure 20: PQRI classification — percentage of ASes in each class on May 1, 2002.

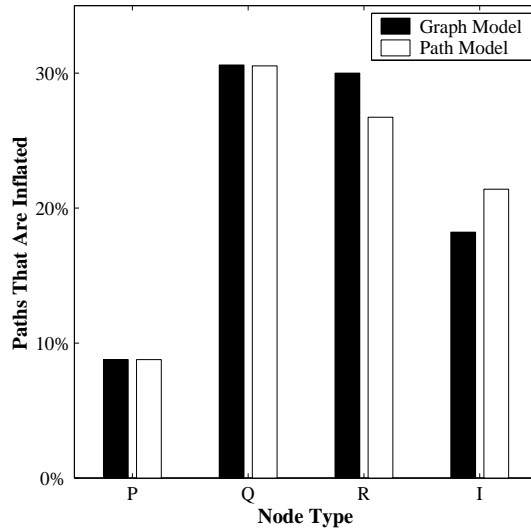


Figure 21: Path inflation by node type on May 1, 2002.

connected to the Internet in terms of number of links, but these links are not as optimal as those they provide to their customers (P- and I-nodes).

We also considered the length of inflated paths in our analysis and saw that there is no correlation between the length of inflated paths and node classification. While there are more inflated paths for R- and Q-nodes, they do not derive more from an optimal route than those of P- and I-nodes.

Summing up our findings on AS classification, we see that looking at the path model gives us a more exact view of the distribution of different node types. It is remarkable that half of all ASes are multi-homed stubs. As mentioned in section 4.1, route aggregation may influence classification results since an AS, which seems to not forward traffic, may be a relay station for hidden ASes.

5 Robustness

More and more people and institutions make use of the multifaceted possibilities of the Internet and therefore depend increasingly on the faultless working of the worldwide net. Investigation of the Internet's robustness hence plays an important role when attempting to develop realistic models and to find weaknesses of the current system.

5.1 Fundamental Considerations

The results in section 4 show that negligence of routing policies may yield falsified or even wrong beliefs about the Internet's structure. Thus, it would be careless not to take policies into account in the case of robustness investigation. When looking at graphs, the number of edge- and node-disjoint paths are useful measures. Another possible approach not included here would be investigating the diameter.

The number of edge-disjoint paths between two nodes states how many connections exist that have no edge, i.e. link, in common while the number of node-disjoint paths gives the same for connections with no common node, i.e. host, router, or AS, respectively.

Since we use AS level models, the nodes represent ASes and the edges existing connections between them. "Existing connections" means that — contrary to a router level model — we know neither the precise number of them nor between which stations they are exactly since an AS may possibly include several BGP routers and gateways. Additionally, the throughput of these links is not known. These are the reason why we restrict to the consideration of *node*-disjoint paths, in the following.

In order to determine the number of disjoint paths, we are looking once more at both representations — graph and paths in the database — separately to gain insight into what influence and consequences policies have. The graph model represents how ASes are connected with each other while the path model considers policies and shows the extent of their influence. However, BGP routing tables do not have to and often do not list all possible (disjoint) paths in their routing tables.

As start node we choose the AS of the University of Oregon since all paths begin there, while the end nodes correspond to the destination ASes found in the BGP routing tables. The following section presents the used algorithms.

5.2 Algorithms

5.2.1 Graph Model

The task to determine the number of *edge*-disjoint paths between two nodes in a graph may be easily reduced to calculating the maximum flow between those nodes where all edges are of weight 1. However, we aim to calculate the number of *node*-disjoint paths. A possible way would be to determine the minimum number of nodes that have to be removed from the graph to interrupt the connection between source and target node. Since we have to check *all* possibilities, this procedure is quite inefficient and may take days to finish on a workstation.

A much better way to get the desired number of disjoint paths is to replace each node by two nodes — an entry node and an exit node — and a connecting edge of weight 1 in between as illustrated in figure 22.

In that way, we get a new graph in which each path through a specific node must traverse the connecting edge. We may now apply a maximum flow algorithm on the resulting graph to

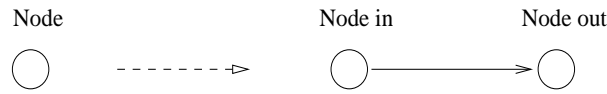


Figure 22: Duplication of nodes in order to efficiently calculate the number of node-disjoint paths.

determine the number of node-disjoint paths. The upper part of figure 23 shows a simple graph whose nodes are connected in only one direction for clarity. The graph has a maximum flow of 3, i.e. 3 edge-disjoint paths, and 2 disjoint paths. The lower part shows the transformed graph where each node is replaced by an entry and exit node connected through an edge. When determining the maximum flow of this new graph we get a value of 2 which matches the number of disjoint paths in the original graph.

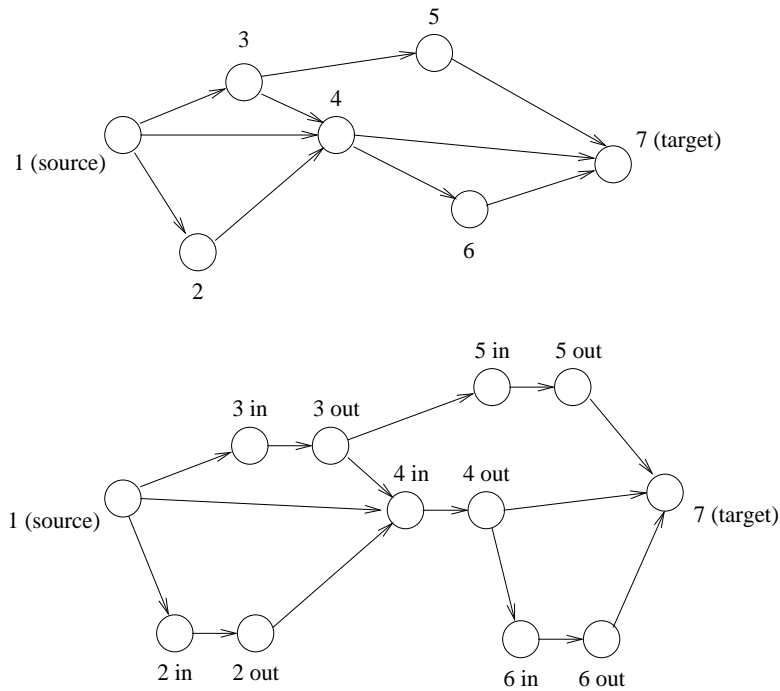


Figure 23: The number of node-disjoint paths in the top graph coincides with the maximum flow (i.e. number of edge-disjoint paths) in the bottom graph.

5.2.2 Path Model

The algorithm described above is based on the graph and therefore may not take the consequences of policies into account. We have seen in section 3.2.1 that new paths may arise in a graph that are not listed in the BGP routing table and hence are not valid according to policies. Thus, the number of disjoint paths in the graph provides an upper bound of the number of really existing disjoint paths. In the following, we present the different algorithms we use to calculate the number of disjoint paths on the basis of the information in our database.

As explained in section 3.2.2, each path is stored along with the corresponding IP prefix. Results in section 4 have shown that for the same destination AS different active paths may exist since one AS often owns multiple IP prefixes. For our analysis, we do not care about IP prefixes. We simply create a list of all differing paths (not only best routes, but *all* paths) and calculate the number of disjoint paths to each destination AS.

The algorithms are performed *for each destination separately*. A first approach is as follows:

1. Remove the path containing the AS number which occurs most frequently in all paths. If multiple paths apply, take the one which intersects the most other paths, i.e. has AS numbers in common with the most other paths.
2. Repeat step 1 until no path is left having a multiply occurring AS number.
3. The number of remaining paths corresponds to the number of node-disjoint paths.

This algorithm looks quite simple, but we have to apply it to the paths stored in the database. The following enumeration shows how our implementation looks like. For better understanding, we describe the steps with words rather than giving the specific SQL statements.

1. For each path, create a list of AS numbers occurring in the path.
2. Create a list of AS numbers that appear in more than one path and store their number of occurrences. If there are no such ASes left go to step 6.
3. Create a list of paths that contain one or more of the multiply occurring AS numbers determined in step 2 and calculate the number of such ASes, for each path.
4. Remove the path with the most occurrences of multiply occurring AS numbers. If multiple paths apply, remove the one with the node appearing in the most paths.
5. Go back to step 2.
6. Determine the number of node-disjoint paths by counting the remaining paths in the list.

The procedure may be done for all destinations simultaneously when grouping by destination AS. In practice, we restricted to about 100 at the same time to get optimal overall performance.

The algorithm described above is *heuristic* and may thus provide results which are not optimal, i.e. there may be more disjoint paths to some destinations than actually found. It runs in $O(n^2 * l)$ (???) where n is the number of paths and l is the length of the longest path. When looking at the problem on a more abstract level one notices that it is equivalent to the task of determining the maximum number of disjoint sets out of some given set. This problem is known to be NP-hard and hence may not be solved in polynomial time. Therefore, it is a good idea to try a second heuristic method to check the solutions from the first one.

The classical approach to get a solution for the disjoint sets problem is as follows:

1. Take the path that intersects fewest other paths. If multiple paths apply take the one which has fewest multiply occurring AS numbers.

2. Remove all paths that intersect the path chosen in step 1.
3. If there are intersecting paths left go to step 1.
4. The number of remaining paths corresponds to the number of node-disjoint paths.

Compared to the first one, this algorithm runs faster since in step 2 all intersecting paths are removed instead of only one at a time like in step 1 of the first approach, but the worst case is still $O(n^2 * l)$ (???). The SQL implementation may be reused with two exceptions: the first value to sort by is the number of intersecting paths instead of the number of appearances of multiply occurring AS numbers in a path which becomes the second value to sort by. Additionally, the sorting order is ascending in order to get the path intersecting *fewest* other paths.

First, we apply the former algorithm on all data in our database. Then we use the latter method only for the paths with destination ASes for which the former algorithm provided a value below the upper bound, namely the number of disjoint paths in the graph.

Totally, the database contains about 155,500 entries of ASes (13 BGP routing tables) for which the number of disjoint paths has to be determined. For about 120 of them, the second algorithm yields better results than the first one, whereas the first algorithm surpasses the second one for about 15 ASes. This might be because both algorithms do not treat the case where more than one path could be chosen in a step. They greedily take the first one instead of trying all possibilities. For each AS, we store the better value from the two algorithms in the database.

The above results lead to the following conclusions:

- None of the presented algorithms gives the optimal, i.e. correct, result in every case.
- Overall, the second procedure provides marginally better results. Only a very small fraction of ASes is affected, though (less than 0.08%).
- It may be assumed that the gained values are reliable enough for our analysis although there are probably a few ASes for which better values may be found.

A possible improvement of this algorithm might be to make allowance for the case that in step 1 several paths may have a minimum number of multiply occurring ASes. In this case, all possibilities should be tried. However, we decided not to implement this because only very few changes may be expected.

5.3 Evaluation of Results

By means of the obtained results we aim now to investigate the following questions:

- How does the number of disjoint paths evolve over time in the graph and the path model?
- How distinct is the discrepancy between the values determined on the basis of the graph and the path model?
- How do the differences between the various structural types (PQRI) look like?

- How are the relations between the degree of a node and the number of disjoint paths to it?

Results of PQRI classification in section 4.2.2 show that on May 1, 2002, 32.6% of all ASes were P-nodes. P-nodes are leaves by definition and hence are always reachable through only one disjoint path in both the graph and path model. In the following, we are going to *ignore P-nodes* to improve clarity of results.

5.3.1 Development over Time

The development over time may be treated in short. Investigations about path inflation and structural issues in section 4 have already shown that temporal changes have become inessential in the considered time period. The same observation holds also for the number of disjoint paths, namely there are hardly any fluctuations ($\pm 0.5\%$). Thus, we are not going into details in this context, but concentrate on the values of May 1, 2002 for our further analysis.

5.3.2 Graph and Path Model

In order to get an image of the number of disjoint paths to Q-, R-, and I-nodes and its distribution, we look at the histogram in figure 24. It shows the values for the graph model. Take into account that the y-axis is logarithmic. About 90% of ASes have 5 or less disjoint paths, and only 1% of ASes is reachable over one single disjoint path. There is another striking point: 74 ASes with more than 35 disjoint paths exist.

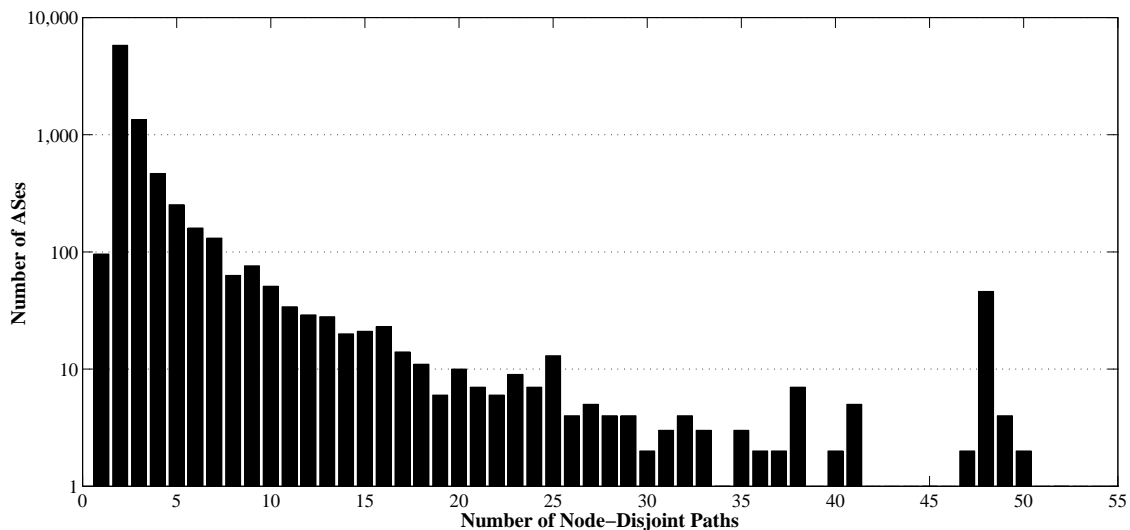


Figure 24: Distribution of the number of node-disjoint paths in the *graph model*.

Before we explore and interpret these values we compare them with the number of disjoint paths in the path model, pictured in figure 25. In accordance with our expectations, even more ASes have 5 or less disjoint paths ($\approx 95\%$) and there are no ASes with more than 35 disjoint paths.

The two figures do not allow us to identify the difference between the two models for a dedicated AS. Therefore, we provide another histogram (see figure 26) showing how the

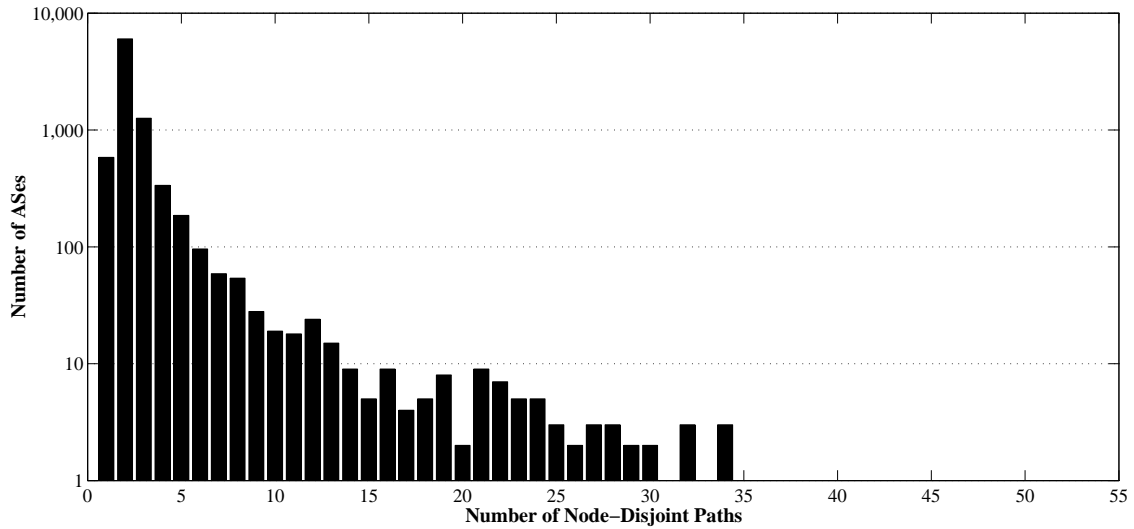


Figure 25: Distribution of the number of node-disjoint paths in the *path model*.

difference between the two models is distributed. One interesting insight is that 20% of Q-, R-, and I-ASes have a different number of disjoint paths in the two models. What is surprising is the fact that 71 ASes have a difference of more than 20 paths. Although this corresponds to only 0.8% of all non-P-ASes, we are going to see in section 5.3.4 that a particular reason for this big discrepancy exists.

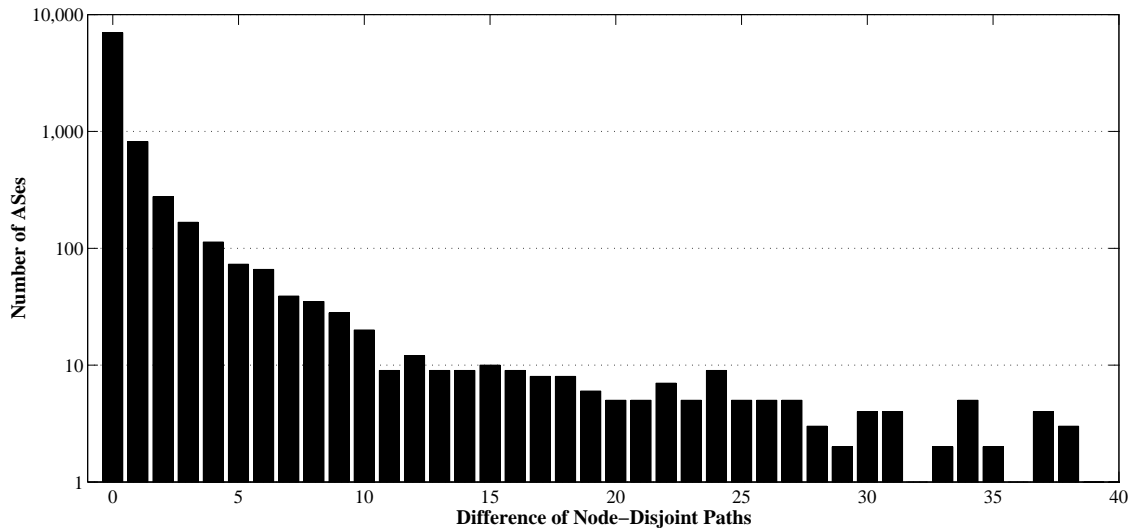


Figure 26: Distribution of the difference of number of node-disjoint paths between graph and path model.

Next, we are going to dwell on the question which relations may be observed between the number of disjoint paths and the structural type of a destination AS.

5.3.3 Structure Influence

By determining the PQRI type in the graph and path model, we obtain a simple but valuable classification of ASes with respect to their role in the Internet. Let us first look at figure 27. It shows the average number of disjoint paths in both models. The node type classification refers to the definition in the path model.

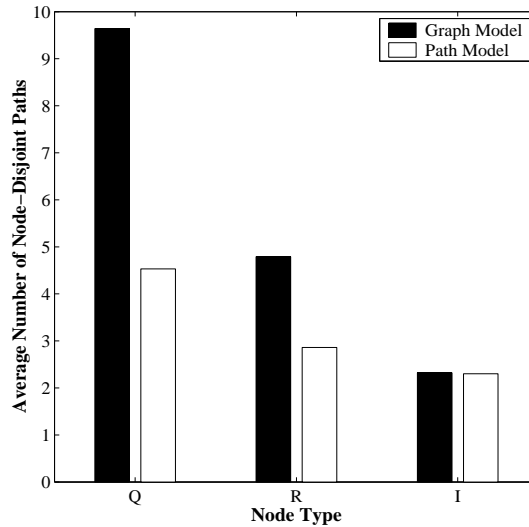


Figure 27: Average number of node-disjoint paths in both models (for node types according to the *path model*).

An insight we may obtain from figure 27 is that Q-nodes have the most disjoint paths on average. In fact, three quarters of all ASes with more than 10 disjoint paths are Q-nodes. Figure 29 and 30 in the appendix show the distribution of the number of disjoint paths more detailed.

The number of disjoint paths to a node is associated with its degree. Namely, the number of incident edges is an upper bound for the number of disjoint paths to an AS. Table 2 shows the average in-degree in our bidirected graph, grouped by node type. Truly, Q-nodes have the biggest degree on average, followed by R-nodes. We discuss this relation further in section 5.3.4.

Node Type	\emptyset Degree	\emptyset Disjoint Paths
Q	28.45	9.64
R	5.34	4.79
I	2.33	2.32

Table 2: Average in-degree and number of node-disjoint paths in bidirected graph, per node type.

We should now address the difference between the average in the graph and the path model for Q- and also R-nodes. Figure 28 indicates the percentage of ASes with differing number of disjoint paths in the graph and path model. The white bars show the percentage

of ASes for the classification defined in the path model. Striking is the difference to the original classification (graph model) for R-nodes. 60% of the R-nodes in the graph model are solely final destinations and thus belong to the I-group in the path model. Hence, the R-nodes in the path model act also as relay stations. This corresponds to the function of Q-nodes and explains the higher percentage of differing numbers for the path model classification.

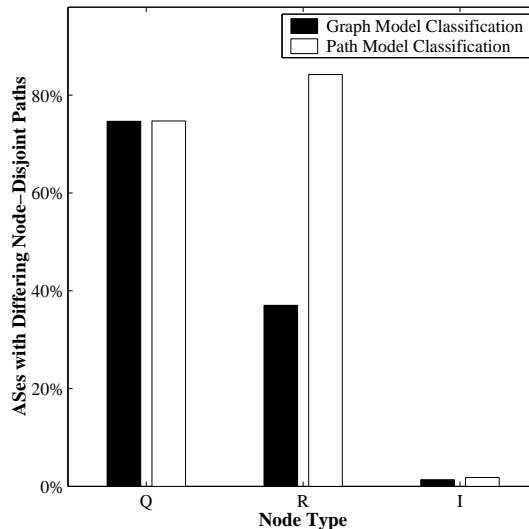


Figure 28: Percentage of ASes with differing number of node-disjoint paths in the two models (for both classifications).

It may be seen that 75% of all Q-nodes and 85% of all R-nodes (considering the path model classification) have different number of disjoint paths in the two models. In other words, a lot of Q- and R-nodes are *physically* well connected and reachable through many disjoint paths (bigger number of disjoint paths in the *graph model*), but not all of these possible paths are listed in the BGP routing table.

We discuss this phenomenon by means of three aspects:

- Q-nodes have high degrees in the bidirected graph. A significant part of links leads to customers (P- and I-nodes). When looking at the really existing paths in the routing table, we find that much more edges are outbound links than inbound. This point is investigated in section 5.3.4.
- Q- and R-nodes are physically well connected, but a considerable number of the theoretically possible paths are not used due to policies.² It is not in the interest of a provider to pay too many core ASes for connections and traffic relaying, but balance “connectivity and budget”. Not listed paths may appear as soon as a corresponding agreement has been concluded by the parties because of increased traffic or closer collaboration.
- Q-nodes are providers and their high degree points out the good connectivity to several core networks to guarantee reliable service. A significant part of the traffic is only

²This becomes also manifest in the case that certain links between ASes are used in one direction only although bidirectional use would improve connectivity.

transiting and not destined for the AS itself. The BGP routing tables however provide information about reachability of an AS as a *final* destination. Paths that are used when the AS acts as a relay station may not be the same as when it is a final destination. Similar argumentation holds for R-nodes which build the core network of the Internet.

The last two points are assumptions. The second last is discussed in section 5.3.4, the last in section 5.3.5.

5.3.4 Number of Disjoint Paths and Degree

Let us take a closer look at the relationship between the number of disjoint paths and the degree of an AS. Formally expressed, we may put it into the form

$$\text{degree} \geq (\# \text{ of node-disjoint paths, graph model}) \geq (\# \text{ of node-disjoint paths, path model})$$

Table 2 indicates the in-degree for the different types in a bidirected graph, along with the number of disjoint paths. Striking is that the difference between the two values is significantly bigger for Q-nodes than for R- and I-nodes. The values in table 3 help us to understand this. There, the number of in- and outbound links is given obtained by looking at the really existing paths in the routing table. Additionally, the number of disjoint paths in the path model is specified.

Node Type	\emptyset Links Out	\emptyset Links In	\emptyset Disjoint Paths
Q	24.56	4.99	4.53
R	2.66	3.00	2.86
I	0.00	2.33	2.30

Table 3: Average number of out- and inbound links and number of node-disjoint paths in the *path model*.

Q-nodes have almost five times more outbound links than inbound. This results from the role of the Q-nodes: they are typically big providers which connect several P- and I-nodes to the Internet and also relay traffic to other Q- and R-nodes. The number of disjoint paths depends on the number of links that are usable for inbound traffic which is not the case for connections to P- and I-nodes. Hence, we have a big difference between the in-degree and the number of disjoint paths for Q-nodes in table 2. For R-nodes, however, the numbers for in- and outbound links differ much less: they do not have any P-nodes and only a few I-nodes as customers. Mainly, they connect different core-networks and relay traffic between them.

Now, we address ourselves to the issue of the differences between disjoint paths in the two models. The number of inbound links to a node gives an upper bound for the number of disjoint paths to this node in the path model. We see in table 3 that, also for Q-nodes, the differences are not too big. Table 2 shows that — in the graph model — 9.64 disjoint paths³ lead to a Q-node on average, but we find only 4.99 links on average entering a Q-node when looking in the routing table. Hence, there are at least about 5 links more which could be used for inbound traffic, but are only listed as outbound links in the routing table. With

³The number of disjoint paths to a node serves as a lower bound for the number of inbound links to this node.

other words, the number of possible inbound links is halved, possibly because current policies do not allow the use of these links for entering traffic.

The same may be observed in a less distinct form regarding R-nodes. 3.00 links are listed for inbound traffic in the routing table while at least 4.79 links could be used according to graph results.

The given explanations base on average values, table 4 shows specific values for three existing ASes. Alternet and Swisscom are Q-nodes and Stanford belongs to the R-group.

Alternet is the most connected AS according to our data. While it has links to 2594 other ASes, only 54 of them are listed for incoming traffic in the routing table. The Swisscom AS is reachable through 48 disjoint paths in the graph whereas policies do not allow more than 38 paths. Actually, the AS is reachable through 32 paths only. The Stanford AS has more incoming links than outgoing ones, and a disjoint path leads through each of the 8 incident links.

AS	Degree	ND Graph	Links Out	Links In	ND Path
Alternet (701)	2594	50	2593	54	32
Swisscom (3303)	88	48	53	38	32
Stanford (32)	12	12	4	8	8

Table 4: Example ASes — “Degree” refers to the in-degree in the bidirected graph, “ND Graph” means number of node-disjoint paths in the graph model, “ND Path” the same for the path model.

The results of this section confirm our assumption that a considerable part of possible paths are not listed in BGP routing tables. This may be caused by policies which result in worse reachability of some Q- and R-nodes than what would be possible.

5.3.5 Partial Paths

The last of the two given assumptions explaining the big fraction of Q- and R-nodes with differing number of disjoint paths in the two models assumes that some paths to an AS are not used if it is a *final* destination, but if the AS acts as a relay station and thus some paths continue from this AS. In this section, we look at paths ending at customer networks (P- and I-nodes) connected to a Q- or R-node, respectively. We call the path from the start AS to the last Q- or R-node *partial path*.

To verify that partial paths exist that do not match a path listed in the routing table, we modify the paths in our database: if the last AS number in a path is a P- or I-node (in accordance with the *path* model classification) it is removed. That way, we get paths which all end at a Q- or R-node. One has to get straight that the obtained paths are *not* valid in terms of policies since a path is only valid for the given last AS number as a destination and not for intermediately occurring ASes. Nevertheless, this modification is very helpful to find out how many disjoint paths exist to a Q- or R-node independently of whether it is the final destination or not. The determination of the number of disjoint paths passes identically to the description in section 5.2.2.

The obtained results are simple and surprising at the same time. The number of disjoint paths for Q- and R-nodes and thus the percentage of ASes with differing number of disjoint paths in the graph and path model remains almost the same. This contradicts the assumption

made above. For a big fraction of Q- and R-nodes, not all physically available disjoint paths through these nodes are listed in the BGP routing table. The observation fortifies once more the first assumption that policies influence the number of disjoint paths to an AS. However, one must not forget that a BGP routing table does not necessarily contain all valid paths to an AS.

5.3.6 Provider Core

In section 5.3.4, we met two ASes with more than 35 disjoint paths in the graph. Figure 24 identifies 74 such ASes and table 5 lists four of them. With the exception of eight R-nodes, these ASes are all Q-nodes. Striking is their number of connections to customers. Truly, they have so many that they are providers of 57% of all links from Q- to P- and I-nodes in the Internet. Because they account for only 6.4% of all Q-nodes, it is plausible that they — without the R-nodes — build something like the “provider core”.

AS	Degree	ND Graph	# of Customers
Alternet (701)	2594	50	2186
Qwest (209)	782	48	601
Time Warner (4323)	202	48	165
Swisscom (3303)	88	48	44

Table 5: Examples of main providers and the number of P- and I-nodes they are connected to.

Reinforced is the above theory when looking at big customers. A few of them are listed in table 6. All of them mainly rely on this provider core. Microsoft’s MSN uses 24 of these providers and three R-nodes. Google uses only these providers. A further reason to choose these ASes is that they provide an optimum of node-disjoint paths.

AS	Degree	ND Path
Microsoft MSN (12076)	27	27
Telocity (12050)	23	23
Google (15169)	11	11

Table 6: Examples of highly connected I-nodes.

Consequently, the provider core is crucial for the overall functioning of the Internet. A simulation where we assumed that it failed disconnected 46% of all customer ASes while Q- and R-nodes were affected by only 15%.

Summarizing our findings about node-disjoint paths we may say that most Q-, R-, and I-nodes are reachable through more than one disjoint path. Primarily Q-nodes, but also R-nodes, have high numbers of disjoint paths in the graph, but a lot of ASes are not reachable through all of these paths when looking at the routing table. Secondly, there is some kind of a provider core which strikes with 35 to 52 disjoint paths in the graph model and maintains more than half of all links to customer ASes.

6 Conclusion

We presented the impact that BGP routing policies have on path inflation and robustness of the Internet. We used two different approaches to represent the Internet on the AS level. The graph model includes all ASes as nodes and links between ASes as bidirectional edges. The resulting graph shows an Internet, as it could be if no BGP routing policies were applied and all links were used in both directions. In the path model, existing paths to all destination ASes are stored in a database to reflect policies. In order to see the extent of policies, results of database queries were compared to those of algorithms run in the graph.

We have got surprising as well as “to-be-expected” results. More important, they point out the strengths and weaknesses of the two models.

First, let us mention that path inflation depends very much on data sets used and on the extracted information. Different ASes maintain different BGP routing tables where path inflation varies from 15 to 50% when extracting the longest best route and from 5 to 40% for the shortest best route. Whether extracting the longest or the shortest or any other route returns meaningful results, remains an open question. Furthermore, a comparison of AS hops is not always accurate and looking at the router-level may provide additional insight. Nevertheless, path inflation is a proven fact.

In the context of path inflation, we observed that the AS of the University of Oregon has a broad view of the Internet and that it is well suited as starting point when investigating BGP routing policies. Since we chose to restrain our work to a single vantage point of the Internet, Oregon is the best possible provider for BGP routing information.

As [VHE01] stated, more and more ASes in the Internet are multi-homed stubs (I-nodes) because of an increasing demand for access to multiple ISPs. With the help of the path model, more than half of all ASes were identified as multi-homed stubs in May 2002. Classifying ASes generally, a weakness of the graph model was revealed; it may not identify all ASes correctly.

Further classification results include an indication that R- and Q-destinations have more inflated paths than Is and Ps. This shows that providers (Q-nodes) and core ASes (R- and Q-nodes) have more differentiated and complicated routing policies than customers. Considering the length of inflated paths, no correlation to node classification could be pointed out.

An insight we obtained while studying the robustness of the Internet is that Q-nodes are reachable through the most node-disjoint paths on average. In fact, three quarters of all ASes with more than 10 disjoint paths are Q-nodes. In direct correlation to this, we found that Q-nodes are of the highest degree. The relationship of disjoint paths to an AS and its degree confirms our assumption that a considerable part of possible paths are not listed in BGP routing tables. The graph model shows that policies result in non-optimal reachability of some Q- and R-nodes.

Finally, the graph model indicates that there may be a group of ASes bordered by their high number of disjoint paths. It includes mostly Q- and a few R-nodes. We identified the Q-nodes as the “provider core” of the Internet since they maintain more than half of all links to customer ASes. They are crucial to keep the Internet connected.

What remains is that, while every BGP routing table is only a snapshot of the Internet, the work presented here, too, discusses momentary findings. Investigations about path inflation and structural issues over a time period of one year did not show any remarkable trends. The same holds true for the number of disjoint paths, namely that there have been hardly any fluctuations since May 2001. Whether the future will bring significant changes or not and how the discovered “provider core” will develop, must be left to further studies.

A Appendix

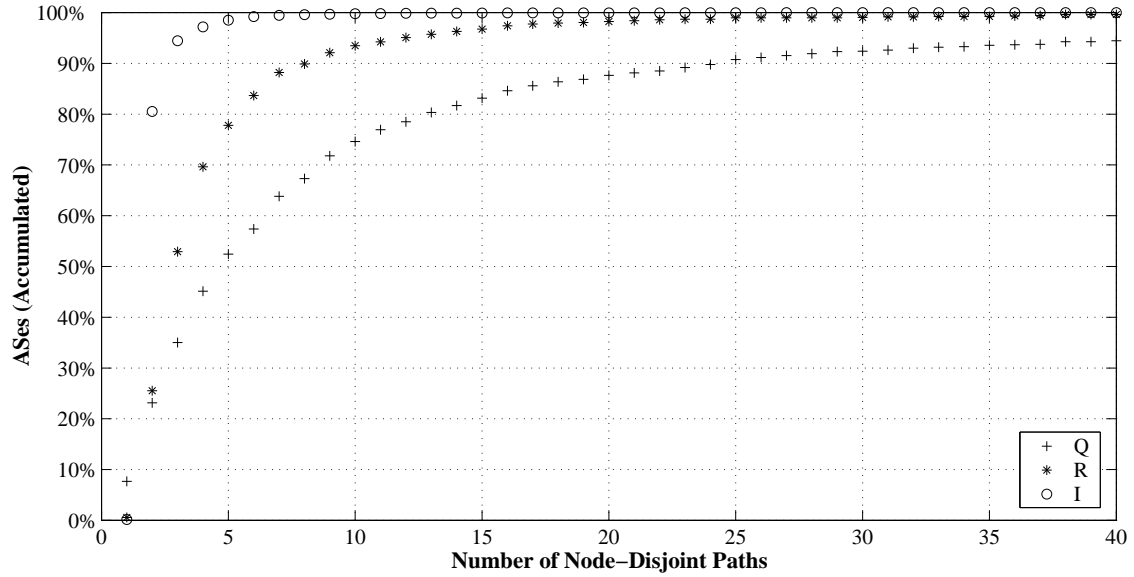


Figure 29: Accumulated percentage of ASes with a specific number of node-disjoint paths in the *graph model* — example: 52% of all Q-nodes are reachable through 5 or less disjoint paths and 70% of R-nodes through 4 or less.

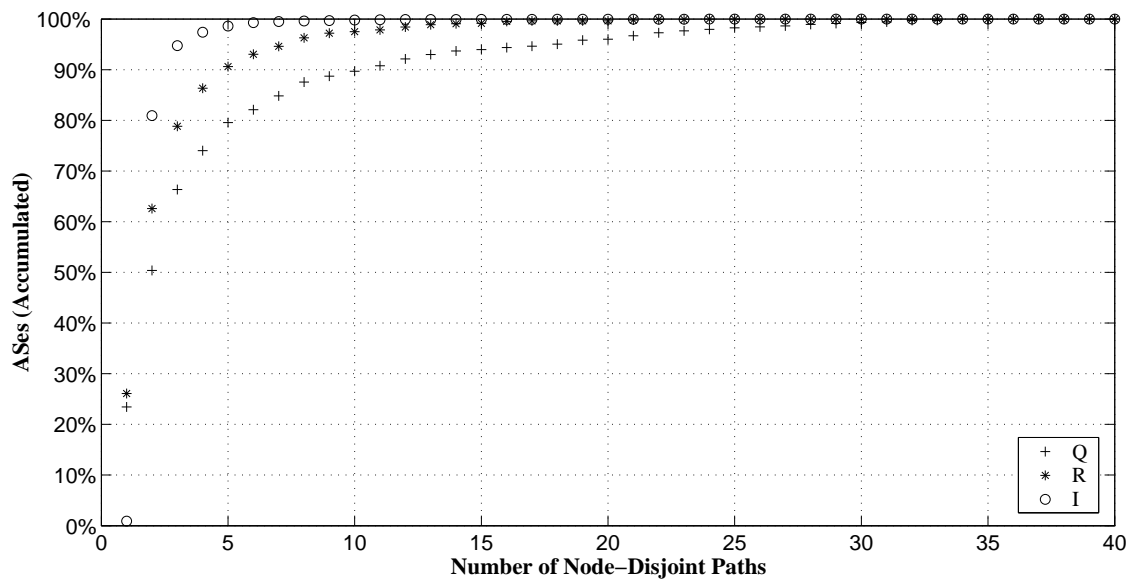


Figure 30: Accumulated percentage of ASes with a specific number of node-disjoint paths in the *path model*.

References

- [Aga] Sharad Agarwal. Characterizing the Internet Hierarchy from Multiple Vantage Points. <http://www.cs.berkeley.edu/~sagarwal/research/BGP-hierarchy>.
- [Alg] Algorithmic Solutions — LEDA Library. http://www.algorithmic-solutions.com/as_html/products/leda.
- [Cisa] Cisco Systems — Understanding Route Aggregation in BGP. <http://www.cisco.com/warp/public/459/aggregation.html>.
- [Cisb] Cisco Systems — BGP Best Path Selection Algorithm. <http://www.cisco.com/warp/public/459/25.shtml>.
- [FBR01] Nick Feamster, Jay Borkenhagen, and Jennifer Rexford. Controlling the Impact of BGP Policy Changes on IP Traffic. *AT&T Labs Technical Memorandum*, November 2001.
- [Fix] Fixed Orbit — Network Search. <http://www.fixedorbit.com/search.htm>.
- [Gao01] Lixin Gao. On Inferring Autonomous System Relationships in the Internet. *IEEE Global Internet*, November 2001.
- [GW01] Lixin Gao and Feng Wang. The Extent of AS Path Inflation by Routing Policies, 2001.
- [HB96] J. Hawkinson and T. Bates. RFC 1930, Guidelines for creation, selection, and registration of an Autonomous System (AS), March 1996.
- [Hus] Geoff Huston. Internet BGP Table. <http://bgp.potaroo.net>.
- [Jun] Juniper Networks — How the Active Route Is Determined. <http://arachne3.juniper.net/techpubs/software/junos50/>.
- [Mey] David Meyer. University of Oregon Route Views Project. <http://www.routeviews.org>.
- [MSOP99] D. Meyer, J. Schmitz, C. Orange, and M. Prior. Using RPSL in Practice, August 1999.
- [MyS] MySQL — Open Source Database. <http://www.mysql.com>.
- [RL95] Y. Rekhter and T. Li. RFC 1771, A Border Gateway Protocol 4 (BGP-4), March 1995.
- [SARK01] Lakshminarayanan Subramanian, Sharad Agarwal, Jennifer Rexford, and Randy H. Katz. Characterizing the Internet Hierarchy from Multiple Vantage Points. *UC Berkeley Technical Report*, August 2001.
- [TGS01] Hongsuda Tangmunarunkit, Ramesh Govindan, and Scott Shenker. Internet Path Inflation Due to Policy Routing, 2001.

- [TGSE01] Hongsuda Tangmunarunkit, Ramesh Govindan, Scott Shenker, and Deborah Estrin. The Impact of Policy Routing on Internet Paths, 2001.
- [VHE01] Danica Vukadinović, Polly Huang, and Thomas Erlebach. A Spectral Analysis of the Internet Topology, July 2001.