# ANALYSIS OF OVERLAY-UNDERLAY TOPOLOGY CORRELATION USING VISUALIZATION

Vinay Aggarwal Anja Feldmann
*Deutsche Telekom Laboratories*
*Berlin, Germany*
*{vinay.aggarwal, anja.feldmann}@telekom.de*

Marco Gaertler Robert Görke Dorothea Wagner
*Universität Karlsruhe (TH)*
*Karlsruhe, Germany*
*{gaertler, rgoerke, wagner}@ira.uka.de*

**ABSTRACT**

In the design and implementation of the overlay architecture most peer-to-peer (P2P) systems rely on the underlay network to provide them with basic connectivity. Therefore, the intrinsic features of the underlay network determine the efficiency of the overlay. Accordingly, studying the interdependency of the overlay and underlay networks leads to a better understanding of P2P behaviour. We present a visualization-driven analysis for evaluating the overlay architecture with respect to the underlay. Using Gnutella as a case study, our analysis confirms that Gnutella's topology differs from a randomly generated network and that there is an implicit correlation between the overlay and underlay topologies.

**KEYWORDS**

Gnutella, Random Network Topology, Autonomous System Graph, Overlay-Underlay Correlation, Core, Visualization

## 1. INTRODUCTION

The recent growth of P2P file sharing applications with respect to total Internet traffic (Sen and Wang '02, Karagiannis et al '04) has led to an unprecedented interest in their analysis (e.g. Aggarwal et al '04). The P2P file sharing applications create overlay networks on top of the Internet underlay. The contemporary Internet is a collection of segregated routing domains called Autonomous Systems (AS), each having their own administration and independent routing policies. Routing information between ASes is exchanged via an exterior gateway protocol such as BGP (BGP). The AS network possesses an implicit hierarchical structure where the ASes can be categorized into backbones, national, regional or local providers, and customers (Gao, L. '00). The graph of the ASes, where nodes represent different ASes, and edges correspond to traffic trade agreements between the ASes, provides us with an abstraction of the Internet underlay. Accordingly, we can say that the P2P file sharing systems create overlay topologies on top of the AS graph. When constructing their topology by forming neighborhoods, most structured and unstructured P2P protocols do not explicitly take the underlay routing into account. Yet as the underlay connectivity determines the overlay performance, understanding the correlation between routing in the overlay and underlay layers is crucial.

There have been some investigations on such correlations recently. Using game theoretic models, Liu et al '05 study the interaction between overlay routing and traffic engineering within an AS. Ratnasamy et al '02 present a node-partitioning scheme that allows overlay nodes to choose peers that are relatively close in terms of network latency. An analysis of routing around link failures (Seetharaman, S. and Ammar, M. '06) finds that tuning underlay routing parameters improves overlay performance. Most investigations tend to point out that the overlay topology does not appear to be correlated with the underlay (e.g., Aggarwal et al '04), but the routing dynamics of the underlay do affect the overlay in ways not yet well understood.

In this paper, we analyze the correlation between overlay and underlay topologies of the Gnutella network (Gnutellav0.6) using a visualization-driven approach (Baur et al '04) which relies on the concept of cores. We confirm that the topology of Gnutella is not uniformly random. We also observe a correlation between features of Gnutella and the AS graph.

The paper is organized as follows. In Section 2, we introduce the experimental setup for Gnutella network measurement and the visualization technique. Section 3 details the overlay-underlay correlation analysis and the comparison of overlay topology with random networks. In Section 4, we make a case for the feasibility of our approach by examining the potential bias in our analysis, and conclude in Section 5.

# 2. PRELIMINARIES

In this section, we introduce the methodology we deployed to collect Gnutella traces, and the visualization technique used to correlate the Gnutella topology with the Internet underlay.

## 2.1 Gnutella network measurement

Gnutella is a popular file-sharing network with roughly 2 million users (Slyck), and has attracted a healthy interest from researchers, e.g., (Stutzbach et al '05, Ripeanu et al '02). The Gnutella (Gnutella v0.4) network is comprised of agents called servents, who can initiate as well as serve requests for resources. When launched, a servent searches for other peers to connect to by sending Hello-like `Ping` messages. The `Pings` are answered by `Pong` messages, which contain address and shared resource information. Search queries are flooded within the Gnutella network using `Query` messages, and answered by `Query Hits`. To limit flooding Gnutella uses TTL (time to live) and message IDs. Each answer message (`Query Hit/Pong`) traverses the reverse path of the corresponding trigger message. While the negotiation traffic is carried within the set of connected Gnutella nodes, the actual data exchange of resources takes place outside the Gnutella network, using the standard HTTP protocol. Due to scalability problems, later versions of Gnutella (Gnutella v0.6) introduced a hierarchy which elevates some servents to ultrapeers, while others become leaf nodes. Each leaf node connects to a small number of ultrapeers while each ultrapeer maintains a large number of neighbors, both ultrapeers and leafs. To further improve performance and to discourage abuse, the `Ping/Pong` protocol underwent semantic changes. Answers to `Pings` are cached (Pong caching) and too frequent `Pings` or repeated `Querys` may cause termination of connection.

In order to analyze the overlay structure, we first need to identify a representative set of edges in the P2P network. By an edge, we mean a direct P2P connection between two overlay nodes. The most obvious way of finding edges in a P2P network is to create some by participating. Yet these are not representative as they are highly biased by the location and software of the participant. Rather we wish to identify edges in the P2P network where none of the two end-nodes is controlled by us. We call such nodes remote neighbor servents.

Due to Pong caching and the rapid fluctuation in Gnutella networks (we measured the median incoming/outgoing connection duration to be 0.75/0.98 seconds), the simple crawling approach using crawler `Pings` (`Pings` with TTL 2) employed in Saroiu et al '02 will likely result in servents that are not active any more. They should, however, have been remote neighbor servents at some point.

To cope with these complications, we deploy a combination of active and passive techniques to explore the Gnutella network. Our passive approach consists of an ultrapeer based on the GTK-Gnutella (GTK-Gnutella) program. The goal is to have an ultrapeer that behaves like a normal node in the network, yet worthwhile to connect to. It shares 100 randomly generated music files (totalling 300 MB in size) and maintains 60 simultaneous connections to other servents. To derive various statistics the servent is instrumented to log per-connection information augmented with a packet level trace. The passive approach gives us a list of active servents.

The active approach consists of a multiple-client crawler that uses `Pings` with TTL 2 to obtain a list of candidate servents. Since `Query` results are difficult to cache, we use `Querys` with TTL 2 to obtain a set of remote neighbor servents. These servents are then contacted actively to further advance the network exploration. This approach allows us to discover Gnutella edges that existed at a very recent point of time. When interacting with other servents, our crawler pretends to be a long-running ultrapeer with a non-provocative querying scheme. It processes incoming messages and has a non-intrusive `Ping/Pong` behaviour. For instance, the servent issues `Query`/crawler `Pings` only to those peers that have already responded with a `Pong`, `Pings` are issued only to those servents that send one themselves. This pragmatic behaviour seems to avoid bans. The client uses `Query` messages with catchwords like mp3, avi, rar. One can expect `Querys` to yield only a subset of neighbors due to the presence of free-riders. We combine active and passive approaches by integrating the crawler into the ultrapeer. Experiments with the unmodified and modified ultrapeer confirm that the changes do not alter the characteristics of incoming connections.

Using this setup, we collect Gnutella logs for one week starting April 14th, 2005. During this time, the ultrapeer logged 352,396 sessions and the crawler discovered 234,984 remote neighbor servents.

## 2.2 Visualization technique

The concept of *cores* (Batagelj V. and Zaversnik M. '02, Seidmann, S. '83) describes a hierarchical decomposition of the nodes of a graph. More precisely, the *k-core* of an undirected graph is defined as the unique subgraph obtained by recursively removing all nodes of degree less than k. A node has *coreness* α, if it belongs to the α-core but not to the (α+1)-core. The *α-shell* is the collection of all nodes having coreness α. The *core* of a graph is the k-core such that the (k+1)-core is empty. Generally the core decomposition of a graph results in disconnected sub-graphs, but in the case of the AS graph we observe that all k-cores stay connected, which is a good feature regarding connectivity. Cores have been frequently used for network analysis, e.g., Gkantsidis et al '03, Gaertler M. and Patrignani M. '04.

Baur et al '04 proposed a visualization technique for drawing the AS network based on a hierarchical decomposition. More precisely, their algorithm incrementally layouts the graph starting from the innermost shell and then iteratively adds the lower shells. Their implementation uses core decomposition and a combination of spectral and force-directed layout techniques. A successful application of this visualization technique compared actual AS graphs with generated AS graphs. The obtained layouts clearly revealed structural differences between the networks.

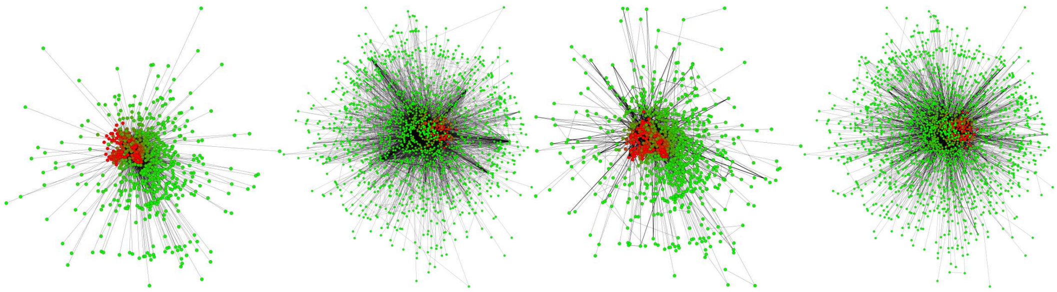
# 3. EVALUATION OF OVERLAY-UNDERLAY CORRELATION

Overlays are formed at the application layer, but the actual data flow takes place at the network layer. Since the neighborhood selection process of overlay networks is largely arbitrary, it becomes interesting to analyze how much the neighborhood selection process of P2P protocols respects the underlying Internet topology. For this purpose, we employ the visualization technique in Section 2.2 to compare the graph imposed by the P2P network to a randomly generated one at the level of the AS graph.

We formalize the task of comparing two overlays as the comparison of two tuples (O_1, U_1) and (O_2, U_2), where O_i represents the overlays and U_i the underlays, while i=1 denotes Gnutella and i=2 denotes the random network. Since the networks U_1 and U_2 need not be the same, e.g., sampled at different points in time or location, these four graphs have to be related to each other. Our approach provides a potential solution by mixing information of the overlay and the corresponding underlay. More precisely, the overlay is reduced to a composition of elements in the underlay.

To derive the graph imposed by the P2P network we utilize the data from the Gnutella measurement setup in Section 2.1. For each edge of the Gnutella network we map the IP addresses of the Gnutella peers to ASes using the BGP table dumps offered by Routeviews (Routeviews) during the week of April 14, 2005. This results in 2964 unique AS edges involving 754 ASes, after duplicate elimination and ignoring P2P edges inside an AS. For the random graph we pick end-points at the IP level by randomly choosing two IP addresses from the whole IP space. These edges are then mapped to ASes in the same manner as for the Gnutella edges. This results in 4975 unique edges involving 2095 ASes for the random network at the AS graph level. The different sizes of the graphs are a result of the generation process: we generated the same number of IP pairs for random network as observed in the Gnutella sample, and applied the same mapping technique to both sets, which abstracts the graph consisting of IPs and direct communication, to a graph with ASes as nodes and the likely underlay communication path. This way, the characteristics of Gnutella are better reflected than by directly generating a random AS network of the same size as Gnutella network.

In the following, we consider two kinds of abstractions of the (communication) paths induced by a communicating pair in the overlay: *direct overlay communication* and *induced underlay communication*. In direct overlay communication, we consider an AS-abstracted view of the direct P2P communication graph, where nodes are ASes and an edge connects two nodes if there exists a direct P2P communication between the corresponding ASes. We define the *appearance weight* of an edge as the number of such communications between the corresponding ASes. Note that the edges in this model disregard the underlying topology. For induced underlay communication, we associate each overlay edge with the corresponding underlay path. This path is computed by building an AS graph from the AS path information from the Routeview data sets and then extracting the AS-neighbor information and locating a likely path for the P2P communication. We remove all edges of the underlay graph that are not used in any overlay communication. The *appearance weight* of an edge denotes the number of paths it appears in. This refines the meaning of the edges in the original underlay network i.e., an edge is present between two ASes if and only if they have a traffic exchange agreement and a P2P communication is routed through it.



(a) Gnutella (b) random network (c) Gnutella (d) random network

Direct overlay communication Induced underlay communication

Figure 1: Comparison of occuring communication in the Gnutella network and a randomly generated network, using visualization, see Section 2.2.

Figure 1 shows a top-down view of the visualizations of communication edges in Gnutella and random network. The visualization technique places nodes with dense neighborhoods (tier-1 and tier-2 ASes) towards the center, and nodes with lesser degrees (tier-3 customer ASes) towards the periphery. We can observe that while both networks have many nodes with large degrees in the center, the random network possesses several nodes with large degree in the periphery. Gnutella, on the other hand, has almost no nodes with large degree in the periphery in both models. Moreover, this pattern is more pronounced for Gnutella in the direct overlay communication model, while the random network is largely similar in both models. In other words, it appears that Gnutella peering connections tend to lie in ASes in the core of the Internet where there may be high-bandwidth links available.

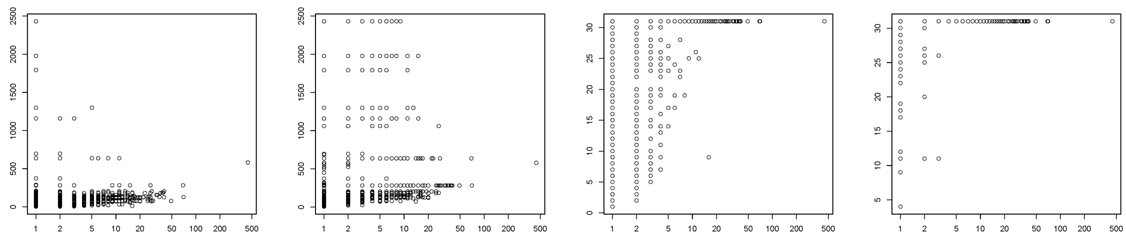
The results from visualizations generally tend to be indicators, which need to be verified by a mathematical analysis. Consequently, to corroborate our observations, we investigate structural dependencies between the induced underlay communication model and the actual underlay network itself. Edges in the underlay network are not equally loaded as some edges appear in more communication paths than others. As it is not possible to measure the actual traffic on the individual edges, we consider a simplified model where a single communication causes one unit of traffic to be routed. The appearance weight of an edge in the underlay communication model thus corresponds to its load. The real load of an edge in the underlay network (including all the traffic caused by other applications) is naturally larger. Comparing these two loads reveals whether the P2P communication has characteristics similar to the accumulated load. This helps in understanding and enhancing the underlay network topology and application level routing techniques. However, measuring the traffic load in the underlay network is not trivial. Even in a simplified model where we consider the load to be equal to the number of appearances in router-path announcements, the measurement is biased. Hence, we compare the appearance weight with node-structural properties of the corresponding end-nodes in the original underlay. We focus on the properties degree and coreness, as both have been successfully applied for the extraction of customer-provider relationship as well as visualization (Subramanian et al ‘04,Gaertler M. and Patrignani M. ’04), as these properties reflect the importance of ASes. We systematically compare the weight of an edge with the minimum and maximum degree and coreness of its end-nodes. Figure 2 shows the corresponding plots.

From the plots of min- and max-degree, it is apparent that the appearance weight of an edge and its end-nodes' degrees are not correlated in Gnutella or the random network, as no pattern is observable. Also, the distributions are similar as the majority of edges are located in the periphery of the network where the maximum degree of the end-nodes is small. We thus hypothesize that the relation of load in the P2P network and node degree in the underlying network is the same in Gnutella and the random network. In other words, the Gnutella network does not appear to be significantly affected by the node degree of underlay nodes.

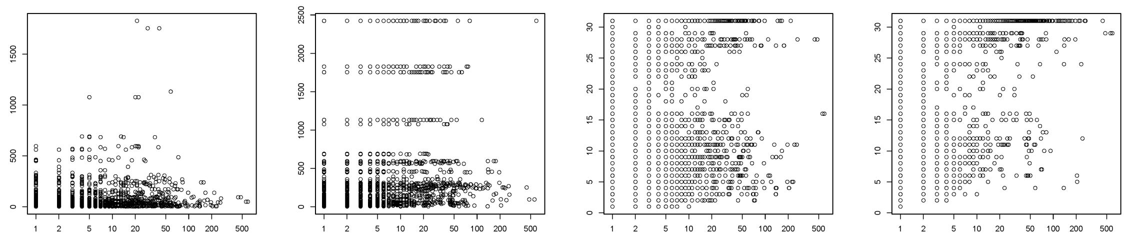
However, considering the coreness reveals interesting observations. From the graphs of minimum and maximum coreness in Figure 2, we can observe that although there is no correlation in either of the two networks, their distributions are different. In the random network the distributions are very uniform, which is a reflection of its random nature. But in the case of Gnutella almost no heavy edge is incident to a node with small coreness, as can be seen in the minimum-coreness diagram. Positively speaking, most edges with large appearance weights are incident to nodes with large minimum coreness. Interpreting coreness as importance of an AS, these Gnutella edges are located in the backbone of the Internet, an important observation. The same diagram for the random network does not yield a similar significant distribution, thus denying a comparable interpretation. For instance, in the random network, there exist edges located in the periphery that are heavily loaded. As an aside, backbone edges need not necessarily be heavily loaded in either network.

All these observations and analysis show that the Gnutella network differs from random networks and there appears to be some correlation of Gnutella topology with the Internet underlay.

min-degree max-degree min-coreness max-coreness



(a) Gnutella



(b) Random network

Figure 2: Comparing appearance weight with minimum and maximum degree and coreness of the corresponding end-nodes in Gnutella and the random network. Each data point represents an edge, while the x-axis denotes the appearance weight and the y-axis reflects the degrees (coreness) of the end-nodes. All axes use logarithmic scale.

# 4. EXAMINATION OF POTENTIAL BIASES

The comparison of overlay and underlay can be affected by AS data (Routeviews). For example, the standard communication path between two ASes may be unavailable due to maintenance or hardware failures. In the following, we consider an alternative data source (DIMES) as well as several samples of Routeviews data distributed over time. This will show if our analysis is biased by time or source of AS data. We found that our analysis is affected by neither, besides this revealed interesting information about properties of AS graph.

## 4.1 Comparing of different AS Data Sources

The Oregon Routeviews Project is one of the major repositories for snapshots of the AS network using looking glasses. In contrast, DIMES extracts AS relations by traceroute experiments. Comparing the Routeviews AS topological map with that of DIMES, we observe that neither DIMES nor the combination of DIMES and Routeviews result in different visualizations (Figure 3). Different greys correspond to the different edge sets, i.e., light grey to DIMES, dark grey to Routeviews and black to the intersection.

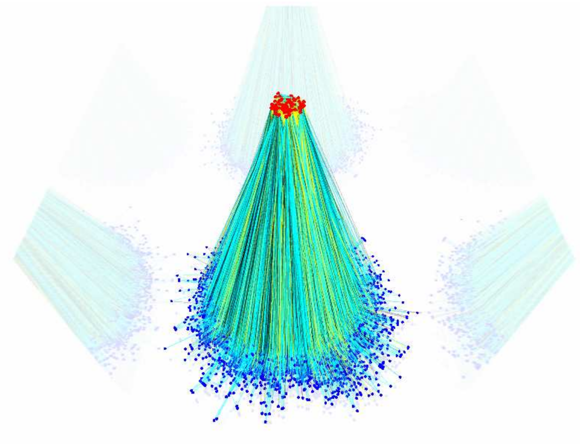
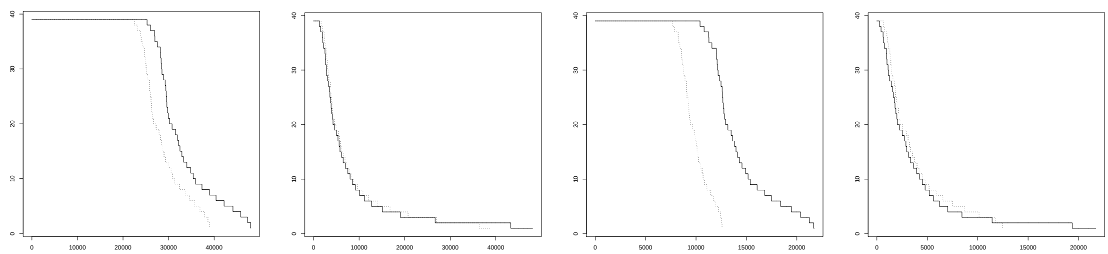


Figure 3: Visualization of the union of the Routeviews and the DIMES data sets.

The data sets correspond to the period of March to June 2005. We obtain 48,073 edges (corresponding to 20,406 ASes) from Routeviews and 38,928 edges (corresponding to 14,154 ASes) from DIMES. Of these, 21,725 edges exclusively belong to Routeviews, and 12,580 edges exclusively to DIMES. The rest of the edges are common to both data sets. The union of the two data sets thus results in 60,653 unique edges (corresponding to 20,612 ASes). Note that the geometric difference of the two data samples is surprisingly large. In other words, 58% of the edges appear in only one data set. An interesting observation is that many edges only discovered by DIMES are incident to the core. Figure 4 shows the plots of the coreness of the end-nodes (which represent ASes) of the edges versus their rank, positioned in the non-decreasing sorted sequence. The coreness is calculated in the graph that consists of the union of the two data samples. This enables us to set up a less biased comparison. The Routeviews data sample is plotted as a solid line, while the DIMES sample is dotted. Figure 4(a) plots a data point for each edge belonging to Routeviews or DIMES using the maximum coreness of the end-nodes (as y-axis), while Figure 4(b) shows the same scenario using the minimum coreness. A similar comparison is made in Figures 4(c) and 4(d) where common edges are omitted. Thus the solid lines represent the distribution of edges that are exclusively observed by Routeviews, and the dotted lines correspond to the exclusive part of DIMES. In principle, the distributions of Routeviews and DIMES are very similar, except for the broad tail of the Routeviews distribution observed in Figure 4(c), which is an interesting observation requiring further investigation. However the overall similarity of the plots and the resembling visualizations reveal that Routeviews and DIMES data is indeed similar, hence our analysis in Section 3 is unaffected by data source.



(a) maximum coreness (b) minimum coreness (c) maximum coreness (d) minimum coreness

Figure 4: Comparison of coreness distributions of edges. Fig a and b compare Routeviews (solid) with DIMES (dotted), while Fig c and d compare the exclusive sets. X axis: number of edges, Y axis: min or max coreness.

## 4.2 MACROSCOPIC EVOLUTION

To ensure that our analysis of the AS graph structure is not biased by the time of measurement, we analyze the temporal evolution of the AS graph obtained from Routeviews over a longer period of time. We use the graph-theoretical concept of k-cores (Batagelj V. and Zaversnik M. ’02 ,Seidmann, S. ’83) to track the general shape of the AS network over time. As illustrated in Figure 5 the visualization technique relates the coreness of an AS to its position in the layout very well: nodes with large coreness are placed in the center while nodes with small coreness are placed in the periphery (this fact was instrumental in Section 3 analysis).

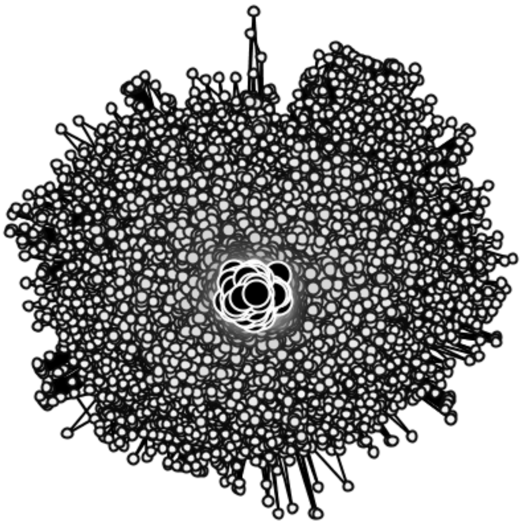


Figure 5: Visualization of the AS network (Jan 1, 2005) using the technique of Baur et.al. '04. Small white nodes have small coreness while big black nodes have large coreness.

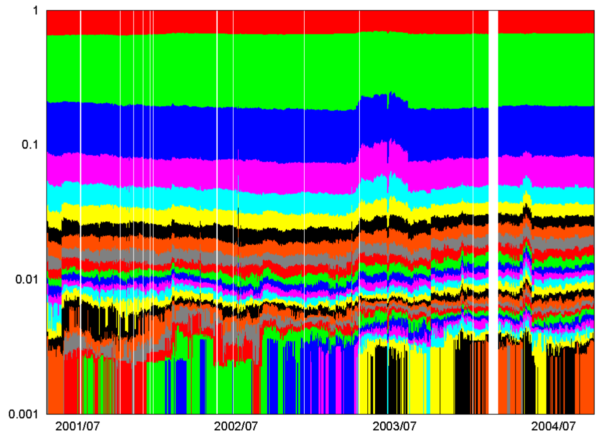



Figure 6: The relative size of shells. The x-axis denotes time and the y-axis (logarithmically scaled) denotes the fraction of nodes in the k-shell. Shells are sorted, with low shells at the top and the maximum at the bottom.

We observe that during the period of April 2001 to April 2005, the number of nodes in the AS graph increases by about 2000 nodes per year, the number of edges increases by 4800 edges per year and the maximum core number has increased from 18 to 26. Although the network grows in absolute terms and especially, the individual core levels grow, their relative sizes remain stable.

Similar to the rings of a tree trunk, Figure 6 illustrates the temporal evolution of the relative proportions of the k-shells, i.e., collection of nodes with coreness k. In this figure, the thickness of one strip corresponds to the fraction of nodes that have a given coreness. The lowest strip represents the maximum core while the highest strip reflects the 1-shell. One can clearly note the stability of k-shells with $k \leq 15$. It is also observable that the size and coreness of the maximum core increases over time. The growth in the coreness is not monotonic and has big fluctuations. The increase is caused by the improving connectivity between major ASes. White vertical strips indicate the absence of data in the collection process. Furthermore, the relative distances of the ASes to the "center" in the visualizations remain roughly the same. The fact that the core structure evolves over time but the relative core sizes remain stable implies that the visualization approach, see Section 2.2 using cores as a means to analyze the AS graph is not biased by the time of measurement. A more detailed analysis of the distribution of the coordinates of the nodes reveals that only 6%-10% are placed in or close to the center. Most nodes having coreness two or three are located in a concentric annulus around the peak. Using several snapshots over time, we found a positive correlation of 0.67-0.78 between the distance from the center and the coreness. This explains the general volcano-like shape reported by Baur et al '04 for the AS graph. This shape also reflects the hierarchical structure of AS graphs well. Furthermore, the shape of the annuli (of the 2- and 3-shell) remains fairly constant over time which indicates the independence of the visualization technique from the size of the graph.

## 5. CONCLUSION

Using visualization and the concept of cores, we have established that while overlay networks like Gnutella use an arbitrary neighborhood selection process, their topology differs from randomly generated networks.

Moreover, there exists some correlation between the overlay and the underlay network topologies. By comparing the Routeviews and DIMES data sets, we confirm that different data sources or collection processes do not significantly affect our view of the AS graph. The analysis of the temporal evolution of the Routeviews data sets shows that the basic structure and properties of the global AS graph remain the same over an extended period of time. Apart from showing that our analysis is not biased by the source of AS data or the time of measurement, this also demonstrates that analyzing the overlay-underlay correlation through a new visualization approach leads to insightful observations. Future work will focus on different overlay networks, characterizing overlay nodes that lie in the core and periphery of visualized graphs using geographic location and uptimes, and augmentations of the visualization-based technique with more elaborate mathematical analysis.

## REFERENCES


Aggarwal et al ’04 (V. Aggarwal, S. Bender, A. Feldmann, and A. Wichmann, 2004). Methodology for Estimating Network Distances of Gnutella Neighbors. In *GI Jahrestagung*, 2004, pages 219--223.

Batagelj V. and Zaversnik M. ‘02. Generalized Cores. *Technical Report 799*, University of Ljublana, 2002.

Battista et al ‘99 (G. Battista, P. Eades, R. Tamassia, and I. Tollis, 1999). *Graph Drawing - Algorithms for the Visualization of Graphs.* Prentice Hall, 1999.

Baur et al ’04 (M. Baur, U. Brandes, M. Gaertler, and D. Wagner, 2004). Drawing the AS Graph in 2.5 Dimensions. *Proceedings of the 12th International Symposium on Graph Drawing (GD'04)*, volume 3383 of *Lecture Notes in Computer Science*, pages 43—48, Springer-Verlag, January 2005.

BGP. *Border gateway protocol*. `http://www.ietf.org/rfc/rfc1771.txt`

Chawathe et al ’03 (Y. Chawathe, S. Ratnasamy, L. Breslau, N. Lanham, and S. Shenker, 2003). Making gnutella-like p2p systems scalable. In *SIGCOMM* 2003.

DIMES. *DIMES - Distributed Internet MEasurements & Simulations*. `http://www.netdimes.org/`

Gaertler M. and Patrignani M. ‘04. Dynamic Analysis of the Autonomous System Graph. In *IPS 2004 - Inter-Domain Performance and Simulation*, pages 13--24, March 2004.

Gao, L. ’00. On inferring autonomous system relationships in the Internet. In *Global Internet Symposium*, 2000.

Gkantsidis et al ‘03 (Gkantsidis, M. Mihail, and E. W. Zegura, 2003). Spectral Analysis of Internet Topologies. In *INFOCOM* 2003

Gnutellav0.4. *Gnutella v0.4.* `http://www9.limewire.com/developer/gnutella_protocol_0.4.pdf/`

Gnutellav0.6. *Gnutella v0.6 RFC.* `http://www.the-gdf.org/`

GTK-Gnutella. *GTK-Gnutella*. `http://gtk-gnutella.sourceforge.net/`

Karagiannis et al ‘04 (T. Karagiannis, A. Broido, N. Brownlee, kc claffy, and M. Faloutsos, 2004). Is p2p dying or just hiding? In *Globecom* 2004.

Liu et al ’05 (Y. Liu, H. Zhang, W. Gong, and D. Towsley, 2005). On the interaction between overlay routing and traffic engineering. In *INFOCOM* 2005.

Ratnasamy et al ‘02 (S. Ratnasamy, M. Handley, R. Karp, and S. Shenker, 2002). Topologically aware overlay construction and server selection. In *INFOCOM* 2002.

Ripeanu et al ’02 (M. Ripeanu, I. Foster, and A. Iamnitchi, 2002). Mapping the gnutella network: Properties of large-scale p2p systems and implications for system design. In *IEEE Internet Computing Journal*, 2002.

Routeviews. *University of Oregon Routeviews Project.* http`://www.routeviews.org/`

Saroiu et al ’02 (S. Saroiu, K. Gummadi, and S. Gribble, 2002). A measurement study of p2p file sharing systems. In *Multimedia Computing and Networking*, 2002.

Seetharaman, S. and Ammar, M. ‘06. On the interaction between dynamic routing in the overlay and native layers. In *INFOCOM* 2006.

Seidmann, S. ‘83. Network Structure and Minimum Degree. In *Social Networks*, 5:269--287, 1983.

Sen and Wang, ‘02. Analyzing Peer-to-Peer Traffic Across Large Networks. In *IMW* 2002.

Slyck. *Slyck*. `http://www.slyck.com/`

Stutzbach et al ’05 (D. Stutzbach, R. Rejaie, and S. Sen, 2005). Characterizing unstructured overlay topologies in modern p2p file-sharing systems. In *IMC* 2005.

Subramanian et al ’04 (L. Subramanian, S. Agarwal, J. Rexford, and R. H. Katz, 2004). Characterizing the Internet Hierarchy from Multiple Vantage Points. In *INFOCOM* 2002.