Julius-Maximilians-Universität Würzburg
Institut für Informatik
Lehrstuhl für Kommunikationsnetze
Prof. Dr.-Ing. P. Tran-Gia

Performance Evaluation and Optimization of Hierarchical Content Delivery Networks

Valentin Burger

Würzburger Beiträge zur
Leistungsbewertung Verteilter Systeme

Bericht 1/16

Würzburger Beiträge zur

Leistungsbewertung Verteilter Systeme

Herausgeber

Prof. Dr.-Ing. P. Tran-Gia
Universität Würzburg
Institut für Informatik
Lehrstuhl für Kommunikationsnetze
Am Hubland
D-97074 Würzburg

Tel.: +49-931-31-86630
Fax.: +49-931-31-86632
email: trangia@informatik.uni-wuerzburg.de

Satz

Reproduktionsfähige Vorlage des Autors.
Gesetzt in LaTeX Linux Libertine 10pt.

ISSN 1432-8801

Performance Evaluation and Optimization of Hierarchical Content Delivery Networks

Dissertation zur Erlangung des
naturwissenschaftlichen Doktorgrades
der Julius–Maximilians–Universität Würzburg

vorgelegt von

Valentin Burger

aus

Kronach

Würzburg 2016

Eingereicht am: 22.01.2016
bei der Fakultät für Mathematik und Informatik
1. Gutachter: Prof. Dr.-Ing. Phuoc Tran-Gia
2. Gutachter: Prof. Dr. Homer Simpson
Tag der mündlichen Prüfung: 20.04.2016

Contents

1 Characterization of the Content Delivery Networks on Autonomous System Level

The performance evaluation of optimization mechanisms in hierarchical content delivery networks requires a thorough knowledge of the nature of current content delivery networks. Such knowledge is required to reflect the characteristics of real-world overlay networks and to define models, parameters and relevant scenarios for performance evaluation. This ensures that the results derived from performance analysis give meaningful insights for real-world scenarios.

The two most popular content delivery concepts are a) peer-to-peer networks (P2P) and b) content delivery networks (CDN). To deliver content, it has to be transported from storage resources to clients. In case of peer-to-peer networks peers contribute storage resources by serving chunks of data. In content delivery networks the storage resources are managed data centers or caches. While in the past peer-to-peer networks where responsible for the most part of traffic, the largest amount of traffic is now transported by CDNs.

In 2015 61% of all Internet video was carried by CDNs according to [19]. The vast amount of traffic causes high costs for ISPs, who need to provide appropriate transport infrastructure and pay transit fees incurred. In order to develop traffic management mechanisms, aiming to reduce transit traffic and optimize content delivery, it is crucial to understand the current situation of content delivery networks and the number and distribution of available resources. The performance of content delivery highly depends on the available resources and the number of consumers.

In this chapter, we characterize the number of resources available in autonomous systems for the two content delivery concepts, peer-to-peer networks and content delivery networks and characterize the topology of autonomous systems.

Due to the location based server assignment for content delivery a distributed measurement architecture is necessary to identify the server resources. While peers can easily be identified by probing trackers, the cache resources of content delivery networks can only be determined by probing the content delivery network from different locations. Typically distributed measurement platforms such as PlanetLab are used for that purpose. The problem is that these measurement platforms may not reflect the perspective as consumer, since they are hosted in NRENs and not in ISP networks. We compare the capabilities of a crowdsourcing platform and a PlanetLab testbed for distributed active measurements.

Recent approaches [20] suggest to use resources on customer premise equipment such as home gateways or network attached storage to support content delivery while saving energy. The potential of such peer assisted content delivery approaches, depends on the number of subscribers in the ISPs network. To get a global view on the number of resources available in each autonomous system we evaluate the Internet Census Dataset, which contains a complete scan of the IP address range.

In order to characterize peer-to-peer networks on autonomous system level, we analyze a measurement study and identify the number of peers per swarm available in each autonomous system and investigate the performance of different peer selection strategies. As key performance indicators we determine *a*) the traffic volume and *b*) the traffic costs of Internet service providers.

The measurement results of the content delivery network YouTube and the characterizations serve as input for the performance evaluation of hierarchical content delivery networks under real world conditions. Therefore, we use them in Chapter **??** to investigate the potential of hierarchical cache networks in realistic parameter studies. In addition our results show that the size of ISP in terms of subscriber is highly heterogeneous, showing that more than 85% of the IPs are active in only 1% of the autonomous systems and that the 10 largest autonomous systems already contain 30% of the active IPs.

The content from this chapter has been published in [21–23]. Section 1.1 describes the background and presents related work. The applied methodology to characterize the BitTorrent traffic and to estimate transit costs is described in Section 1.2. In Section 1.3 two approaches for distributed active measurements are compared. The evaluation of the Internet census dataset is presented in Section 1.4. We discuss lessons learned in Section 1.5.

1.1 Background and Related Work

This section describes background of this chapter and presents related work. We start with an overview on measurement studies of live BitTorrent networks and show different approaches to reduce inter-ISP traffic discussed in the ALTO working group of the IETF. Finally, we introduce studies that infer the inter-AS relations based on BGP routing information. We briefly describe the structure of the YouTube video CDN and give a short introduction in the principles of crowdsourcing. Further, we summarize related work in the field of distributed active measurements of CDNs as well as work related to crowdsourcing aided network measurements.

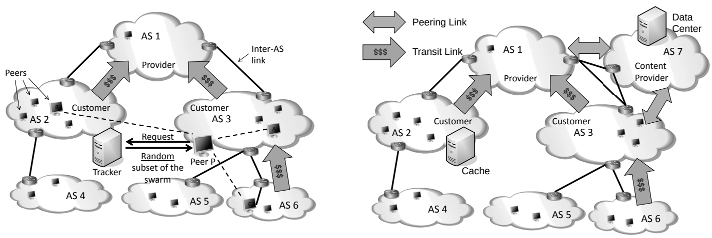
1.1.1 Measurements and Models of Live BitTorrent Networks

BitTorrent is a peer-to-peer file-sharing protocol, which is based on multi-source downloads between the users. All the users, i.e., *peers*, sharing the same file belong to a *swarm*, c.f., Figure 1.1a. To join the swarm, a peer requests addresses of other peers at an index server called *tracker*. In the standard BitTorrent algorithm the tracker uses random peer selection to select a subset of peers that are in the swarm. Then, the joining peer tries to establish a neighbor relation to the peers it got from the tracker and collects all peers which accepted the request in his *neighbor* set. The peer signals interest to all neighbors which have parts of the file it still needs to download. To which neighbor a peer is willing to upload data is decided by the choking algorithm, which is explained in [24].

As basis of our methodology for modeling inter-ISP BitTorrent traffic, the results in [25] are revisited. In [25] the authors provide measurements of a large number of live BitTorrent swarms taken from popular index servers such as *The Pirate Bay*, *Mininova*, and *Demonoid*. Using the IP addresses of the peers, the authors associate every peer with its AS and estimate the potential of ALTO mechanisms based on the differentiation between local peers (peers in the same AS) and remote peers located in other ASes. In contrast, we consider the actual Internet topology in this work, i.e., the inter-ISP relations, the ISP classification in the Internet hierarchy, and the AS paths between the peers in order to estimate the optimization potential of ALTO mechanisms.

The authors of [26] use the peer exchange protocol (PEX) in order to measure the neighbor set of all peers participating in a number of live BitTorrent swarms. Based on this information, they model the graph topology of the swarms and compare the structure to random graphs. They also investigate clustering of peers within ASes and countries, but do not focus on inter-AS relations and AS paths between peers as we do in this work.

In addition, there are measurement studies that examine and model distinct features of BitTorrent networks. In [27], a single swarm was measured for five months with a focus on the download times of the peers. Additional parameters such as the peer inter-arrival times in the swarm, their upload capacity and their online time are considered in [28]. The authors of [29] investigate these parameters also in multi-swarm scenarios. Finally, [30] measures 4.6 million torrents to provide an overview of the entire BitTorrent ecosystem with its different communities and index servers. Our study differs from these works in that it focuses on the location of the peers in the Internet and the AS paths between the peers.



(a) Peer-to-peer

(b) Content delivery network

Figure 1.1: Autonomous system topology with a peer-to-peer (left) and a content delivery network (right).

1.1.2 Measurements of AS Relations and Topologies

Autonomous systems are individual parts of the Internet, which are operated by ISPs. On a technical level, the traffic exchange between the ASes is controlled by the Border Gateway Protocol (BGP)[31]. However, commercial relations between ISPs determine the routing policies configured via BGP. An ISP must buy transit services to access parts of the Internet it neither owns nor can access by its customers. Hence, to route traffic between autonomous systems ISPs engage in business relationships. These business relationships are usually not open for public but they can be abstracted into three common types [32]. The relationship between two ASes can be customer-to-provider (c2p), peer-to-peer (p2p) or sibling-to-sibling (s2s), c.f., Figure 1.1. A customer-to-provider link is present if the customer AS pays the provider AS for transit service, i.e., the provider forwards the traffic of the customer and its customers. In a peer-to-peer relation the ASes have an agreement that they exchange each others traffic and the traffic of their customers, without paying each other. Sibling-to-sibling are links between ASes of the same organization. These relations are defined in business agreements and kept secret, but they can be inferred by analyzing the routing between autonomous systems.

The approach that is most widely used to infer AS relationships is analyzing BGP routing tables. The data set used in this work is also produced by inferring BGP tables as described in [33]. Therefore, AS links are extracted from RouteViews BGP tables. First sibling-to-sibling links are identified by looking up organizations that own multiple AS numbers. Then customer-to-provider relationships are inferred by a heuristic that is based on the idea of relaxing the requirement for a maximal number of valid paths and using the AS degree information to detect paths that are invalid. Most challenging is the inference of peer-to-peer links since paths remain valid if peer-to-peer links are replaced by a customer-to-provider or provider-to-customer link. The authors of [33] develop a heuristic which combines the strengths of previous approaches by [32] and [34]. The inferred relationships were validated by surveys, showing that 96.5 % customer-to-provider, 82.8 % peer-to-peer, and 90.3 % sibling-to-sibling of the inferred relationships are correct.

1.1.3 Evolution and Structure of Content Delivery Networks

Since the launch of the YouTube service content delivery has drastically changed. While the amount of traffic transported over peer-to-peer networks remained about the same, the traffic transported by content delivery networks has increased exponentially [19]. The number of users watching videos on demand has massively increased and the bandwidth to access videos is much higher. Furthermore, the increased bandwidth enables web services to be interactive by using dynamic server- or client-side scripts. The appearance of dynamic services and the increasing quality of multimedia content raised user expectations and the demand on the servers. To bring content in high quality to end-users with low latency and to deal with increasing demand, content providers have to replicate and distribute the content to get it close to end-users. Thus, content delivery networks such as the Google CDN evolved.

The global expansion of the CDNs also changes the structure of the Internet.

Google has set up a global backbone which interconnects Google's data centers to important edge points of presence. Since these points of presence are distributed across the globe, Google can offer direct peering links to access networks with many end users, c.f. Figure 1.1b. Such, access network providers save transit costs, while Google is able to offer services with low latency. To bring content even closer to users ISPs can deploy Google servers inside their own network to serve popular content, including YouTube videos [35].

To select the closest server for a content request and to implement load balancing CDNs use the Domain Name System (DNS). Typically a user watches a YouTube video by visiting a YouTube video URL with a web browser. The browser then contacts the local DNS server to resolve the hostname. Thereafter, the HTTP request is directed to a front end web server that returns an HTML page including URLs for default and fallback video servers. These URLs are again resolved by DNS servers to physical video servers, which stream the content. The last DNS resolution can happen repeatedly until a server with enough capacity is found to serve the request. Thus, load balancing between the servers is achieved [36].

1.1.4 Crowdsourcing

Crowdsourcing is an emerging service in the Internet that enables outsourcing jobs to a large, anonymous crowd of users [37]. So called *Crowdsourcing platforms* act as mediator between the users submitting the tasks, the *employers*, and the users willing to complete these tasks, the *workers*. All interactions between workers and employers are usually managed through these platforms and no direct communication exits, resulting in a very loose worker-employer relationship. The complexity of Crowdsourcing tasks varies between simple transcriptions of single words [38] and even research and development tasks [39]. Usually, the task description are much more fine granular than in comparable forms in traditional work organization [40]. This small task granularity hold in particular for *micro-tasks*, which can be completed within a few seconds to a few minutes. These tasks are usually highly repetitive, e.g., adding textual descriptions to pictures, and are grouped in larger units, so called *campaigns*.

1.1.5 Distributed Measurements of Content Delivery Networks

There already exist a number of publications which study the structure of the YouTube CDN and its selection of video servers. A distributed active measurement platform is necessary for these evaluation, because the CDN mechanisms consider the client locations, both geographical as well as in terms of the connected access network. In [41] two university campus networks and three ISP networks were used to investigate the YouTube CDN from vantage points in three different countries. The results show that locality in terms of latency is not the only factor for video server selection.

While the view of five different ISPs on a global CDN is still narrow, the authors of [42] used PlanetLab to investigate the YouTube server selection strategies and load-balancing. They find that YouTube massively deploys caches in many different locations worldwide, placing them at the edge of the Google autonomous system or even at ISP networks. The work is enhanced in [36], where they uncover a detailed architecture of the YouTube CDN, showing a 3-tier physical video server hierarchy. Furthermore, they identify a layered logical structure in the video server namespace, allowing YouTube to leverage the existing DNS system and the HTTP protocol.

However, to assess the expansion of the whole YouTube CDN and its cache locations in access networks, the PlanetLab platform, which is located solely in National Research and Education Networks (NRENs), is not suitable, since it does not reflect the perspective of end users in ISP access networks. Therefore, a different distributed measurement platform is used in [43] which runs on end user equipment and thus implies a higher diversity of nodes and reflects the perspective of end user in access networks. However, the number of nodes that was available for the measurement is too small to obtain a global coverage of vantage points

To achieve both, the view of access networks and a high global coverage with a large number of measurement points, the participation of a large number of end users in the measurement is necessary. Bischof et al. [44] implemented an approach to gather data form peer-to-peer networks to globally characterize the service quality of ISPs using volunteers.

In contrast to this we propose using a commercial crowdsourcing platform to recruit users running a specially designed measurement software and therewith act as measurement probes. In comparison to other approaches using volunteers, this approach offers better scalability and controllability, because the number and origin of the participants can be adjusted using the recruiting mechanism of the crowdsourcing platform. This is confirmed by Table 1.1 which compares a crowdsourcing study with a social network study quantitatively. The crowdsourcing study is described in [45]. The study is designed to assess the subjective QoE for multimedia applications, like video streaming. The same study was conducted additionally in a social network environment for recruiting test users. Table 1.1 shows that acquiring people in crowdsourcing platforms takes very short time compared to asking volunteers in a social network, which allows adding participants easily. Furthermore, the completion time of the campaign of 31 hours is much shorter compared to the 26 days for the social network campaign. Finally, in the crowdsourcing campaign workers can be selected according to their country, which allows distributing the campaign on many different countries. In the social network the coverage of countries depends on the network of user groups, which spread the campaign. Hence, it is easy to control the number and origin of subjects participating in a crowdsourcing campaign and the completion time is considerably fast, which makes the campaign scalable and controllable. The price you pay is the reward for the workers that summed up to a total of 16 Euro for that campaign.

The Internet Census Dataset was validated forensically in [46]. In [47] the scope of the dataset is taken into perspective and show that, although there are some qualitative problems, the measurement data seems to be authentic. We use

Table 1.1: Quantitative Comparison: Crowdsourcing / Social Network Study.

	Crowdsourcing (C)	**Social network (S)**
Implementation time	about 2 weeks; test implemented via dynamic web pages, application monitoring	same as for (C)
Time for acquiring people	5 minutes	2 hours, as users (groups) were asked individually
Campaign submission cost	16 Euro	0 Euro
Subject's reward	0.15 Euro	0 Euro
Number of test conditions	3	3
Advertised people	100	350
Campaign completion time	31 hours	26 days; strongly depends on advertised user groups however
Participating users	100	95
Reliable users (very strict filtering of users)	30	58
Number of different countries of subjects	30	3; strongly depends on users groups however

the Internet Census Dataset to determine the number of active IP-addresses for each autonomous system in the Internet.

1.2 Traffic Characterization of Peer-to-Peer Networks

The most popular peer-to-peer overlay network today is BitTorrent. BitTorrent is still responsible for a large portion of Internet traffic [19, 48]. In particular, BitTorrent networks generate a lot of inter-ISP traffic, which is often costly for the ISPs. One approach to optimize the traffic flows, which has recently received a lot of attention is Application Layer Traffic Optimization (ALTO), i.e., *P2P guidance*, to increase the efficiency of BitTorrent and to reduce the amount of inter-ISP traffic and costs. Evaluations of such approaches have been conducted mostly in controlled, artificial scenarios. Examples for such scenarios are simulations with homogeneous peer distributions across ISPs, the evaluation of simple topologies, like the star topology with a tier−1 ISP in the center. However, in today's Internet the inter-ISP traffic routing is based on a complex topology defined by inter-ISP relationships, e.g., peering or customer-to-provider, and ISP classifications such as tier−1, large and small ISPs, and stub ISPs. Hence, these economic relations play an important role in the actual Internet traffic flow. However, this topology of the Internet is not taken into account by most evaluations of P2P guidance approaches, which limits the practical relevance of the results. Furthermore, it is an open question how much BitTorrent traffic is located in which region of the Internet. However, this is a prerequisite in order to estimate the potential of ALTO mechanisms.

To model the BitTorrent traffic flow across ISPs in the Internet, we use measurements of live BitTorrent swarms and the actual autonomous system (AS) topology of the Internet provided by Caida.org. Thus, we estimate BitTorrent traffic characteristics and the emerging transit costs. The measurements of live BitTorrent swarms contain the location of peers in the Internet, i.e., AS numbers, for a very large set of swarms [25]. The dataset from Caida.org contains a full AS graph derived from RouteViews BGP table snapshots, including the AS relations. We infer AS paths based on this dataset with the algorithm published in [49]. We use the inferred AS paths to calculate the real AS paths between peers in BitTorrent swarms. In addition, we define three peer selection strategies that decide which peer in a swarm is connected to which other peer: (a) The random selection strategy is applied to peer selection of today's BitTorrent clients. (b) The locality aware selection strategy connects to the peers with shortest AS paths, to reduce AS hops and potentially latencies in the BitTorrent network. (c) The selfish-ISP selection connects peers preferentially to peers in the ISPs customer tree in order to maximize its revenue. The locality aware and selfish-ISP selection strategies are motivated by the optimization potential of the BitTorrent overlay network, as in [25], and the optimization potential of the revenue of ISPs transit services, respectively.

The contribution of this section is two-fold. First, our results show how Bit-

Torrent networks are distributed over the Internet. We find that almost none of the peers are located in tier−1 ASes which means that tier−1 ASes are not able to control BitTorrent swarms by directly accessing the peers. We analyze the amount of BitTorrent traffic each AS forwards with BitTorrent random peer selection. From our results we derive that most traffic is forwarded by tier−1 ASes on a per AS basis, whereas the BitTorrent traffic aggregated over all large ISPs is significantly higher. As second contribution, we analyze the potential to optimize the BitTorrent overlay by using the shortest AS paths. We find that in about 15 % of the investigated swarms, peers exist who can exchange data locally in the same AS. Furthermore, the AS path length has a median of two AS hops for random selection whereas AS paths have two or less AS hops in 80 % with the locality selection strategy. The inter-AS traffic is reduced especially in tier−1 and in large ISPs by locality aware peer selection. Finally, we estimate the potential of ISPs to optimize their revenue. We find that tier−1 ASes loose a lot of revenue if locality or selfish-ISP selection is used because they are avoided as provider. Small ISPs and stub ASes have large benefits from using locality because they can minimize their costs. This implies less revenues at tier−1 and large ISPs. Large ISPs have to use selfish-ISP selection to have a higher prospect on profit.

The applied methodology to characterize BitTorrent traffic and to estimate transit costs is described in Section 1.2.1. In Section 1.2.2 we describe the numerical examples of this study and their importance for ISPs.

1.2.1 Method for Modeling BitTorrent Traffic Flow and Revenue of ISPs

In this section we describe the methodology to estimate transit costs of ASes. First, we show where we obtain the AS affiliation of peers. Second, we explain how AS paths are inferred from AS relations and how to classify the ASes. Further on we describe different BitTorrent peer selection strategies determining the connection among peers in the Internet. Finally we introduce our transit cost model.

AS Affiliation of Peers

In order to know where peers are located and where BitTorrent swarms generate costs for ISPs, we need to know how the swarms are distributed over the Internet and in which ASes the peers are located. For that purpose, we use the dataset of BitTorrent movie torrents "`Mov.`" provided by the authors of [25]. A snapshot of all available movie torrents on Mininova.org was taken. The swarm sizes and peer distributions were recorded by distributed measurements. The data set consists of files with AS number and number of peers pairs for each BitTorrent swarm. Hence, they provide information for each swarm on how many peers are located in which AS. The measurement took part in April 2009 and recorded 126 050 swarms. Peers of 8 492 ASes are present in the swarms.

AS Relations, Paths and Classification

To be able to estimate the transit costs produced by peers exchanging data in BitTorrent swarms, we need to know the AS paths that connect the peers. Datasets with complete AS paths would be very large and are not available to our knowledge. Hence, we infer AS paths from AS relationships. We use the AS relationship dataset from Caida.org [50]. The dataset contains AS links annotated with AS relations. Each file contains a full AS graph derived from RouteViews BGP table snapshots. For our estimations we use the dataset from January 2011. The dataset consists of customer-to-provider and peer-to-peer relations. However, sibling-to-sibling is only considered by Caida and does not occur in the dataset.

We implemented the algorithm described in [49] in Java to infer the AS paths between any two peers based on the AS relationship dataset. The authors developed a breadth first search algorithm which infers shortest paths conforming to the AS path constraints. The algorithm has runtime $O(N \cdot M)$ for finding all pair valid shortest AS paths of the graph, where N is the number of AS relations and M is the number of ASes. The algorithm's input parameter is the source AS α.

Table 1.2: Classification of autonomous systems.

Type	**Classification**	**#ASes**
Tier−1	AS has no providers	11
Large ISP	AS customer tree ≥ 50	337
Small ISP	AS customer tree < 50 and ≥ 5	1770
Stub	AS customer tree < 5	36289

For every destination AS β the algorithm returns a set of paths $\mathbb{P}(\alpha, \beta)$, which connect α and β.

Further on, we want to obtain results dependent on the AS size and type of business. Therefore we classify the ASes into *stub*, *small ISP*, *large ISP* and *tier−1*. For that purpose, we use a dataset from [51], which provides information about the number of customers and providers for each AS number. This dataset is from November 2011 and is used to classify the ASes according to the size of their customer tree. Table 1.2 lists the different AS types and their classification. Tier−1 ASes are the largest ASes building the core of the Internet. Tier−1 ASes do not have providers. In their dataset 11 tier−1 ASes are identified. If an AS has a customer tree that contains at least 50 nodes, it is classified as a large ISP. An AS is classified as small ISP if its customer tree has less than 50 but at least 5 nodes. Most ASes are stub ASes, which have a customer tree that is smaller than 5.

BitTorrent Neighbor Set Creation

The BitTorrent neighbor set of a peer defines the data exchange with other peers in BitTorrent swarms. Neighbors are the peers in the swarm which are connected to a peer. It has to be noted that the measurements in [25] do not reveal the real composition of the neighbor sets of the swarms. Further on, neighbor sets are randomly generated and differ for every peer, which makes them hard to capture. Hence, we estimate the composition of the neighbor sets in three simple ways, *random*, *locality* and *selfish-ISP*. The number of peers in the swarm is the swarm size S. The number of neighbors is denoted as N with $N \leq S-1$. In the standard BitTorrent implementation a client can connect to up to 40 peers, so we set the maximum size of the neighbor set to $N_{max} = 40$. Hence, the neighbor set for each peer in a swarm with size S has size

$$N = min(N_{max}, S-1)\,. \tag{1.1}$$

We add peers to the neighbor set until it contains N neighbors according to the following algorithms.

random In the random selection strategy we add random N peers of the swarm to the neighbor set. In the standard BitTorrent algorithm the selection of neighbors is also random. Hence, with this selection strategy we try to estimate the traffic and costs produced by the standard BitTorrent algorithm.

locality In the locality algorithm we sort the AS paths connecting two peers by the number of AS hops. Then we add the peers according to the sorted set of increasing AS paths until N peers are in the neighbor set. Note, that first the peers located in the same AS, i.e., zero AS hops, are added. This selection algorithm is used to optimize the swarm by minimizing AS hops between peers and thereby potentially reducing latencies. Hence, the motivation for this algorithm is to optimize the swarm from the overlay's point of view. In practice, such a selection could be realized e.g. with an *iTracker* [52], a database which maps IP-addresses to autonomous system numbers, or other ALTO mechanisms.

selfish-ISP The selfish-ISP selection algorithm tries to select as many peers from customer ASes as possible. Until the neighbor set contains N peers it first adds peers from paths starting with provider-to-customer links, then peers of the same AS, then paths starting with peering links and finally customer-to-provider links. This selection algorithm is used to maximize the revenue of ISPs. This is achieved by the selfish-ISP strategy by selecting preferentially peers that

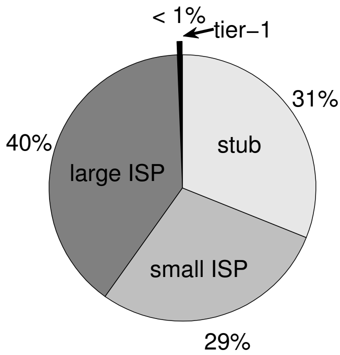


Figure 1.2: Ratio of peers per ISP class aggregated over all swarms in the measurement set.

are connected by customers and avoiding peers at providers. In practice an ISP must be able to control the neighbor set. Hence, ALTO mechanisms for selfish-ISP selection must be controllable by the ISP. One approach is that the ISP provides an information service to guide the peer selection, such as an oracle [53] or an information service [54].

Cost Model

To be able to estimate the costs for ASes arising from transit services, we need to know how much traffic is generated and how much providers charge customers for forwarding the traffic. We consider a snapshot and assume instantaneous traffic rates, i.e., the file-size of the download can be neglected. For simplicity we make assumptions on how much traffic is generated in each swarm, depending on the the number and location of peers.

Assumption 1. *The traffic generated by a peer is equally shared among its neighbors.*

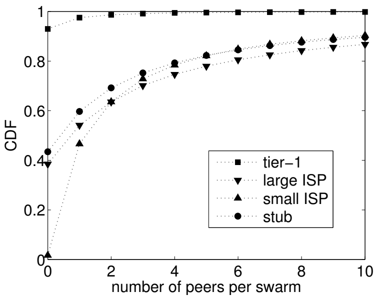


Figure 1.3: CDF of the number of peers per swarm depending on the different ISP classes.

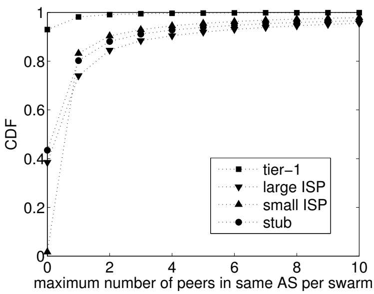


Figure 1.4: CDF of the maximum number of peers in same AS per swarm depending on the tier.

Assumption 2. *All peers generate traffic at the same rate.*

Assumption 3. *The traffic between ASes is equally shared among the paths that connect them.*

In practice, traffic rates are allocated by BitTorrent's choke algorithm, which takes into account the upload and download speed of the other peers. Further on, traffic is generally not shared among different AS paths. But, since we consider the aggregated traffic of a large number of swarms, we argue that these assumptions are reasonable and the results do not change significantly.

Traffic Amount We use the above assumptions to estimate the traffic generated by the BitTorrent swarms. Assumption 1 implies, that the traffic sent by a given peer p_1 is equally distributed among its N neighbors. Hence, the traffic p_1 sends to a neighbor p_2 is

$$T(p_1, p_2) = \frac{1}{N} \,. \tag{1.2}$$

Assumption 2 implies that the traffic originating in a given AS α is proportional to the number of peers located in this AS. Let $\mathcal{S}$ be the set of all swarms, then the traffic of all swarms that is sent from AS α to AS β can be calculated by

$$T(\alpha,\beta) = \sum_{s\in\mathcal{S}} \sum_{\substack{\alpha\in s,\\ p_1\in\alpha}} \sum_{\substack{\beta\in s,\\ p_2\in\beta}} T(p_1,p_2)\,. \tag{1.3}$$

The set of AS paths connecting AS α with β obtained by the AS inference algorithm is given by $\mathbb{P}(\alpha,\beta)$. Assumption 3 implies that the traffic between α and β and later the costs are shared equally among the paths in $\mathbb{P}(\alpha,\beta)$. Hence, we can calculate the traffic on a path $P \in \mathbb{P}(\alpha,\beta)$.

$$T(P) = \frac{1}{|\mathbb{P}(\alpha,\beta)|}\cdot T(\alpha,\beta), \quad P \in \mathbb{P}(\alpha,\beta)\,. \tag{1.4}$$

Next we can calculate the link load $L(\alpha,\beta)$ on the link between two directly connected ASes α and β. We use $\alpha \leftrightarrow \beta \in P$ as notation for a direct link between α and β on the path P. The link load is the sum of the load on all paths sharing the link $\alpha \leftrightarrow \beta$.

$$L(\alpha,\beta) = \sum_{P|\alpha\leftrightarrow\beta\in P} T(P)\,. \tag{1.5}$$

As we consider each AS in a swarm as source AS, the outgoing AS traffic equals the incoming AS traffic. Therefore, we only consider the outgoing AS traffic as inter-AS traffic. The in- and outgoing traffic for AS α is the sum of all loads on links connecting α.

$$in(\alpha) = out(\alpha) = \sum_{\beta|\exists P,\alpha\leftrightarrow\beta\in P} L(\alpha,\beta)\,. \tag{1.6}$$

In the following we estimate the transit costs. The transit costs are weighted by the link loads defined in this section.

Transit Costs

The business relationships between ISPs define the exact transit costs, but they are part of the private contracts between the ISPs. Hence, we develop a simple model for the arising transit costs. It is common that peering ASes exchange their traffic and the traffic of their customers without charging. Hence, we assume no costs for peering links. The amount a customer pays a provider for transit for a specific volume of traffic is unclear, so we set it to one cost unit, i.e., 1. That is not the case in practice, but as we have a large number of ASes and swarms, we get a qualitatively good estimation.
The *costs* of an AS α are increased, if it acts as customer of an AS β. The costs are increased by one unit weighted by the amount of traffic on the link connecting α and β, i.e., $L(\alpha, \beta)$ from Equation 1.5. Let $\mathcal{P}(\alpha)$ be the set of providers of α, and let $\mathcal{C}(\alpha)$ be the set of customers of α. Then we can calculate the costs of AS α emerging in all swarms as follows.

$$costs(\alpha) = \sum_{\beta \in \mathcal{P}(\alpha)} L(\alpha, \beta)\,. \tag{1.7}$$

In the same way we can calculate the *revenues* for all AS links and swarms, where α acts as provider.

$$revenues(\alpha) = \sum_{\beta \in \mathcal{C}(\alpha)} L(\alpha, \beta)\,. \tag{1.8}$$

The *balance* is the difference between revenues and costs.

$$balance(\alpha) = revenues(\alpha) - costs(\alpha)\,. \tag{1.9}$$

1.2.2 Numerical Results and Their Implications

In this section we present the numerical results obtained by applying our methodology to the measurement data and describe their importance for ISPs.

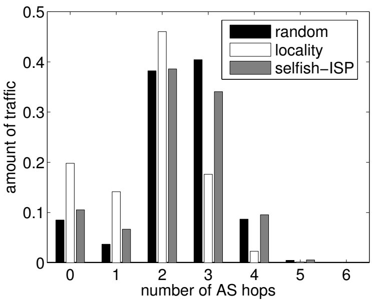


Figure 1.5: Distribution of AS path lengths weighted by the amount of traffic.

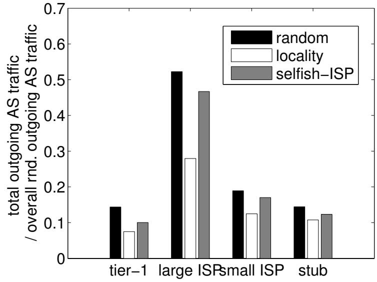


Figure 1.6: Total outgoing AS traffic for different peer selection strategies.

First we show how the peers are distributed over the different ASes. Then we characterize the traffic emerged by BitTorrent swarms and investigate the impact of locality and selfish-ISP peer selection algorithm. Finally, we estimate the transit costs arising by the BitTorrent swarms and investigate the potential of ISPs to maximize their balance by the peer selection algorithms.

Distribution of Peers in the Internet Hierarchy

In the following we describe how peers of BitTorrent swarms are distributed over the Internet. Figure 1.2 shows the distribution of peers over the different tiers. The peers of all torrents are considered. Most of the peers are in large ISP ASes, where 40 % of all peers are located. In small ISP and stub ASes a similar amount of 29 % and 31 % of the peers is located, respectively. Only very few peers are located in tier−1 ASes, which is less than 1 % of all peers in all swarms. Hence, the access to peers by tier−1 ASes is negligible. That means that tier−1 ASes barely have an impact on ALTO mechanisms that control only the peers of the own AS.

Figure 1.3 shows the cumulative distribution function (CDF) of peers per

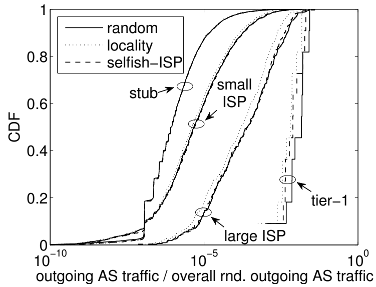
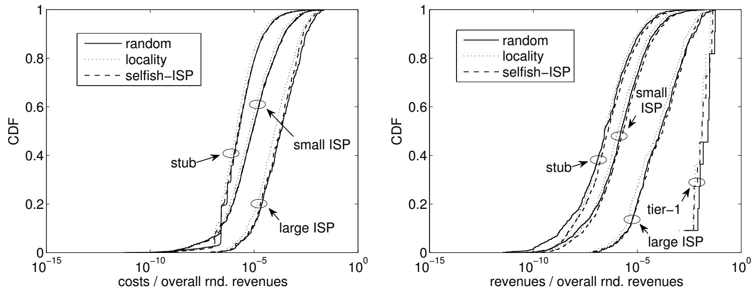


Figure 1.7: CDF of the outgoing AS traffic grouped by AS size.

swarm. We summed up the number of peers being in the same tier for each swarm and calculated the CDF. The probability that at least one peer in a small ISP is existing in a swarm is highest. Only about 2 % of the swarms do not contain any small ISP peer. About 57 % of the swarms contain stub AS peers and more than 60 % contain peers from large ISPs. The probability to find more than one peer of non tier−1 ASes is about 45 % for small ISPs and a bit higher for large ISPs and stub ASes. There are less than 10 % of the swarms which contain a peer from tier−1. Finding more than one peer of tier−1 ASes in one swarm is very unlikely. Few swarms have a very large number of peers, with the maximum of 9 467 peers of one distributed over all large ISPs.

Peers can exchange data locally in the same AS as soon as at least two of them are in it. This cannot be derived from Figure 1.3, because peers can be located in the same tier, but not in the same AS. In Figure 1.4 we calculated the cumulative probability of the maximum number of peers in a swarm which are located in the same AS. As soon as the maximum number of peers in one AS is at least 2, data can be exchanged by the peers locally. Figure 1.4 shows that the probability to exchange traffic locally is low and that large ISPs have the greatest potential. In about 15 % of the large ISPs, peers find neighbors being located in the same AS. For small ISP and stub ASes the chance to find peers of the same swarm in



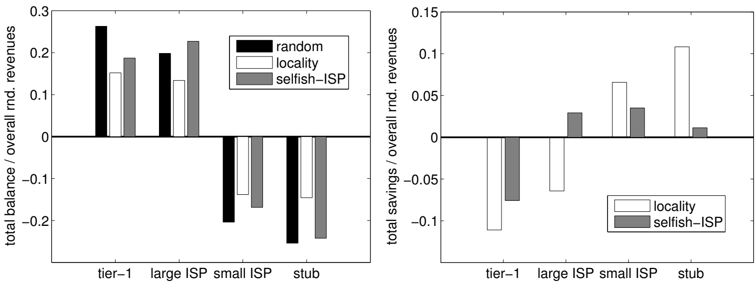
(a) Costs *(b) Revenues*

Figure 1.8: Cumulative probability of transit costs (left) and revenues (right) normalized by the overall revenue for random peer selection grouped by AS size.

the same AS is about 10 % and 12 % respectively. Hence, considering all ASes the potential for local neighbor selection is relatively small, intra-AS traffic is only generated in 15 % and less of non tier−1 ASes. But there are a few swarms with many peers generating a lot of traffic which have a very high potential for traffic optimization. The AS with the most peers in one swarm is a large ISP containing 3 372 peers of one swarm. The dataset contains 42 swarms with more than 1 000 peers in a single AS. In tier−1 ASes there is barely no chance to connect to a local neighbor.

Traffic Characteristics for P2P Guidance Strategies

In this subsection we characterize the traffic produced by BitTorrent swarms. Further on we investigate the potential of ALTO techniques to optimize the swarm in terms of load on the network and AS path length. First we look at the traffic characteristics of the standard BitTorrent algorithm and in the following we compare the different selection strategies. The number of AS hops is the number of ASes on the AS path connecting two peers without regarding the



(a) Balance *(b) Savings*

Figure 1.9: Total balance of transit costs (left) and total savings over random selection (right) normalized by the overall revenue for random peer selection.

source AS. The number of hops are weighted by $L(\alpha, \beta)$, i.e. the amount of traffic and the number of concurring AS paths, see Equation 1.5. Figure 1.5 shows the amount of traffic on AS paths with length in AS hops for the different selection strategies. The median is about 2 AS hops if peers are selected randomly. Most traffic is on paths with two or three AS hops without selection strategy. Paths are up to 10 AS hops in the investigated swarms.

If we use the local selection strategy, the probability for shorter AS paths is higher, compared to random and selfish selection. If local peer selection is used, about 20 % of the traffic can be exchanged in the same AS, i.e. with no AS hop, which is twice as much as for the other strategies. Random and selfish selection have a median of two AS hops, whereas paths have two or less AS hops in about 80 % with local selection strategy. Selfish selection has no considerable potential to reduce the AS path length.

Figure 1.6 shows the amount of inter-AS traffic produced by BitTorrent swarms. We estimate the outgoing traffic of each AS with $out(\alpha)$ in Equation 1.6, i.e. the load produced by all peer-to-peer connections on the links connecting α normalized by the number of neighbors and the number of paths sharing the links. The outgoing traffic of each BitTorrent swarm and each AS is calculated and summed up for the different AS types. For each AS type Figure 1.6 depicts the sum of outgoing traffic normalized by the overall total outgoing traffic produced by random selection of all AS types. The peer selection strategy is coded in the different levels of grey and later line styles. Independent of the selection strategy, most of the traffic is at large ISPs. Less than half of large ISP traffic is at small ISPs. The traffic going out of all the stub ASes is in total a similar amount as the traffic going out of the 11 tier−1 ASes. Hence, most traffic is going out of tier−1 ASes on a per AS basis.

We use the outgoing traffic as a measure for the load on the network. Figure 1.6 depicts the outgoing traffic for the different selection strategies dependent on the AS type. Locality selection reduces the amount of emerging inter-AS traffic in every AS type. Especially large ISPs have a high potential to take load of inter-AS links by selecting local peers. Selfish peer selection reduces the traffic going out of tier−1 ASes, probably because less customers use them as transit providers and route their traffic to customers or keep it local. Apart from that selfish selection does not reduce the load on the network significantly.

Figure 1.7 shows the cumulative distribution function of the outgoing AS traffic grouped by the AS type. The outgoing traffic is normalized by the overall outgoing AS traffic of the random peer selection strategy. AS mention before tier−1 ASes have most outgoing traffic on a per AS basis. Further on we observe that the outgoing traffic decreases with size of the AS. Also noticeable is that with the locality peer selection algorithm we get less outgoing traffic, especially for large ISPs. The difference is not very big for a single AS, but the large number of ASes makes a big difference in the total outgoing AS traffic.

Transit Costs

Now we estimate the transit costs emerged by BitTorrent traffic for the different ISPs and show the potential to save costs and maximize revenues of the peer selection algorithms. We use the overall revenues for random selection, i.e., the sum of total revenues of all AS types, to normalize the values derived in this section. As the overall total balance is zero, the overall total revenues equal the overall total costs. As described in Section 1.2.1 every customer/provider AS α on an AS path connecting peers is charged by $\pm L(\alpha, \beta)$.

Figure 1.8a shows the cumulative distribution function of transit costs, as calculated in Equation 1.7, for the ASes grouped by AS types. Hence, the amount ASes pay providers for transit services. The costs are normalized by the overall revenues of random selection. tier−1 ASes do not have providers and therefore no transit costs. Local peer selection reduces the transit costs, regarding the overall distribution of costs, for all non tier−1 AS types. Costs of large ASes, i.e., ASes that have many customers and forward a lot of traffic, tend be higher.

Figure 1.8b shows the cumulative probability of revenues, see Equation 1.8, of the ASes grouped by AS type. Tier−1 ASes achieve highest revenues. They have the largest customer tree which pay for transit services. ASes with a smaller customer tree get less revenues. The difference between the selection strategies is small for every single AS, but the large number of ASes makes a big difference in the total revenues and further total balance, as we explain in the next paragraph. However, we observe that stub ASes, small and large ISPs tend to have lower revenues using locality selection compared to random selection. In contrast revenues increase with higher probability for selfish-ISP selection in large intervals, in particular from 10^{-8} to 10^{-4} for stub and small ISPs. This was the aim of the selfish-ISP selection strategy. Tier−1 ISPs are loosing revenues if selection strategies are used. Hence, peer-to-peer guidance and selfish-ISP selection are not beneficial for tier−1.

The total balance over all measured BitTorrent swarms is calculated by subtracting costs from revenues of each AS. Figure 1.9a depicts the total balance depending on the AS size. The total balance is normalized by the overall revenues of random selection. The balance is calculated for the standard BitTorrent peer selection, the locality-aware and selfish strategy. For all three strategies, tier−1 and large ISPs have a positive balance and small ISP and stub ASes have a negative balance. This corresponds to the expectation, because tier−1 and large ISPs have many customers whereas small ISPs and stub ASes have many providers. Hence, small ASes have to pay for the transit provided by large ASes.

To highlight the effect of the peer selection strategies on the balance of the ASes, we investigate the savings over random selection. Comparing the local strategy with the standard strategy, we notice that small ASes save costs by selecting local neighbors, resulting in less revenues by the large ASes. Figure 1.9b shows the savings over random selection achieved by using locality and selfish-ISP selection. The savings are calculated by subtracting the total balance with selection strategy from the total balance of the random selection strategy. The savings are normalized by the overall revenues of random selection. tier−1 ASes loose most revenue when local selection is used, which is 10 % of the overall total revenue. The traffic is kept locally and less traffic is forwarded by tier−1 ASes to reach remote destinations. Hence, the transit services of tier−1 ASes are avoided which results in less revenues. Large ISPs also gain less when local peer selection is used. Small ISPs and stub ASes gain from local peer selection because they save costs for transit services by avoiding long AS paths. 10 % of the overall total costs are saved by stub ASes, hence they have the highest potential to profit from selecting peers by locality.

The only way to increase the prospect on higher profit for large ISPs is using the selfish strategy. But also small ISPs have a high potential to maximize their revenues being selfish. Thus, large and small ISPs are in a win-win situation, because they can connect to their plenty customers and do not have to pay for transit services by avoiding connections to providers. This is where tier−1 ASes loose, because less of the ISPs use them as provider in the selfish strategy. Thus, a tier−1 AS cannot be more selfish than in the random selection strategy. Having only few or no customers, stub ASes have poor capabilities to be selfish but avoiding providers also gives them a small advantage over random selection.

1.3 Content Delivery Network Characterization by Distributed Active Measurements

The most popular peer-to-peer overlay network today is Youtube. Therefore we focus on YouTube. Internet video constitutes more than half of all consumer Internet traffic globally, and its percentage will further increase [55]. Most of the video traffic is delivered by content delivery networks (CDNs). Today the world's largest video CDN is YouTube. Since Google took over YouTube in 2006 the infrastructure of the video delivery platform has grown to be a global content delivery network. The global expansion of the CDN was also necessary to cope with growing demand of user demands and the high expectations on the video playback. Therefore, content delivery networks try to bring content geographically close to users. However, the traffic from content delivery networks is highly asymmetric and produces a large amount of costly inter-domain traffic [56]. Especially Internet Service Providers (ISPs) providing access to many end users have problems to deal with the huge amount of traffic originating from YouTube. Furthermore, the Google CDN is constantly growing and changing, which makes it difficult for access providers to adapt their infrastructure accordingly.

To understand and monitor the impact of YouTube traffic on ISPs and the topology of CDNs appropriate measurements are acquired. Due to YouTube's load-balancing and caching mechanisms the YouTube video server selection is highly dependent on the location of the measurement points. Hence, we need a globally distributed measurement platform to perform active measurements to uncover the location of YouTube servers. Recent work [36, 42] has performed such measurements in PlanetLab [57], a global test bed that provides measurement nodes at universities and research institutes. The problem is that probes disseminated from PlanetLab nodes origin solely from National Research and Education Networks (NRENs). This may not reflect the perspective of access ISPs which have a different connection to the YouTube CDN with different peering or transit agreements.

To achieve a better view on the YouTube CDN from the perspective of end users in access networks we use a commercial crowdsourcing platform to recruit regular Internet users as measurement probes. Thus, we increase the coverage of vantage points for the distributed measurement of the YouTube CDN. To evaluate the impact of the measurement platform and the coverage of their vantage point, we perform the same measurements using PlanetLab nodes and crowdsourcing users and compare the obtained results.

Our measurements show that distributed measurements in PlanetLab are not capable to capture a globally distributed network, since the PlanetLab nodes are located in NRENs where the view on the Internet is limited. We demonstrate that recruiting users via crowdsourcing platforms as measurement probes can offer a complementary view on the Internet, since they provide access to real end users devices located out side of these dedicated research networks. This complementary view can help to gain a better understanding of the characteristics of Video CDNs. Concepts like ALTO or economic traffic management (ETM) [**bookchapter2009-12**] need a global view of the CDN structure to optimize traffic beyond the borders of ISPs. Finally, models for simulation and performance evaluation of mechanisms incorporating CDNs need to apply the characteristics identified by crowd sourced network measurements.

The measurements conducted in the PlanetLab and via crowdsourcing are described in Section 1.3.1. In Section 1.3.2 we provide details on the measurement results and their importance for the design of distributed network measurements.

1.3.1 Distributed Active Measurement Description

To assess the capability of crowdsourcing for distributed active measurements we conduct measurements with both PlanetLab and the commercial Crowdsourcing platform Microworkers [58]. We measure the global expansion of the YouTube CDN by resolving physical server IP-addresses for clients in different locations.

Description of the PlanetLab Measurement

PlanetLab is a publicly available test bed, which currently consists of 1173 nodes at 561 sites. The sites are usually located at universities or research institutes. Hence, they are connected to the Internet via NRENs. To conduct a measurement in PlanetLab a slice has to be set up which consists of a set of virtual machines running on different nodes in the PlanetLab test bed. Researchers can then access these slides to install measurement scripts. In our case the measurement script implemented in Java extracted the server hostnames of the page of three predetermined YouTube videos and resolved the IP addresses of the physical video servers. The IP addresses of the PlanetLab clients and the resolved IP addresses of the physical video servers were stored in a database. To be able to investigate locality in the YouTube CDN, the geo-location of servers and clients is necessary. For that purpose the IP addresses were mapped to geographic coordinates with MaxMinds GeoIP database [59]. The measurement was conducted on 220 randomly chosen PlanetLab nodes in March 2012.

Description of the Crowdsourcing Measurement

To measure the topology of the YouTube CDN from an end users point of view who is connected by an ISP network we used the crowdsourcing platform Microworker [58]. The workers were asked to access a web page with an embedded Java application, which automatically conducts client side measurements. These include, among others, the extraction of the default and fallback server URLs from three predetermined YouTube video pages. The extracted URLs were resolved to the physical IP address of the video servers locally on the clients. The IP addresses of video servers and of the workers client were sent to a server which collected all measurements and stored them in a database.

In a first measurement run, in December 2011, 60 different users of Microworkers participated in the measurements. Previous evaluation have shown, that the majority of the platform users is located in Asia [], and accordingly most of the participants of there first campaign were from Bangladesh. In order to obtain wide measurement coverage the number of Asian workers participating in a second measurement campaign, conducted in March 2012, was restricted. In total, 247 workers from 32 different countries, finished the measurements successfully identifying 1592 unique physical YouTube server IP addresses.

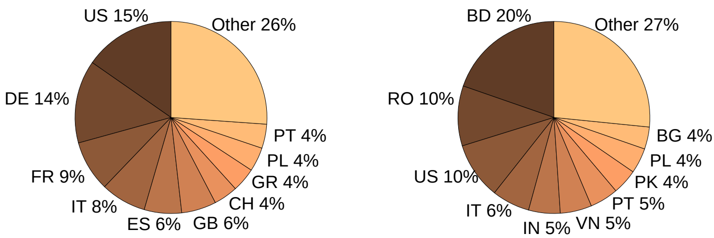
1.3.2 Measurement Results

In this section we show the results of the distributed measurement of the global CDN. The obtained results show the distribution of clients and servers over different countries. Furthermore, the mapping on autonomous systems gives insights to the coverage of the Internet.

Distribution of Vantage Points on Countries

To investigate the coverage of measurement points we study the distribution of the PlanetLab nodes and Crowdsourcing workers. Figure 1.10a shows the distribution of PlanetLab nodes on countries over the world. The pie chart is denoted with the country codes and the percentage of PlanetLab nodes in the respective country. Most of the 220 clients are located in the US with 15% of all clients. However, more than 50% of the clients are located in West-Europe. Only few clients are located in different parts of the world. The tailored distribution towards Western countries is caused by the fact, that the majority of the PlanetLab nodes are located in the US or in western Europe.

Figure 1.10b shows the geo-location of workers on the crowdsourcing platform. In contrast to PlanetLab, most of the 247 measurement points are located in Asia-Pacific and East-Europe. The majority of the participating workers 20% are from Bangladesh followed by Romania and the US with 10%. This bias is caused by the overall worker distribution on the platform []. However, this can be influences to a certain extend by limiting the access to the tasks to certain geographical regions.



(a) PlanetLab *(b) Crowdsourcing*

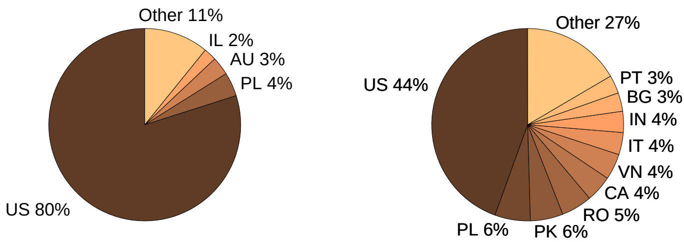
Figure 1.10: Distribution of measurement points on countries in a) PlanetLab and b) Crowdsourcing platform.

Distribution of Identified YouTube Servers on Countries

To investigate the expansion of the YouTube CDN we study the distribution of YouTube servers over the world. Figure 1.11a shows the location of the servers identified by the PlanetLab nodes. The requests are mainly directed to servers in the US. Only 20% of the requests were directed to servers not located in the US.

The servers identified by the crowdsourcing measurement are shown in Figure 1.11b. The amount of requests being directed to servers located in the US is still high. 44% of clients were directed to the US. However, in this case the amount of requests resolved to servers outside the US is higher. In contrast to the PlanetLab measurement many requests are served locally in the countries of clients. Furthermore, the decrease of 80% to 44% of request being directed to the US shows a huge difference.

Hence, network probes being overrepresented in the US and Europe leads to a limited view of the content delivery network and the Internet. This shows the impact of different locations of measurement points on the view of the CDN. It also demands a careful choice of vantage points for a proper design of exper-



(a) PlanetLab *(b) Crowdsourcing*

Figure 1.11: Distribution of physical YouTube servers on countries accessed from a) PlanetLab nodes and b) workers of a crowdsourcing platform.

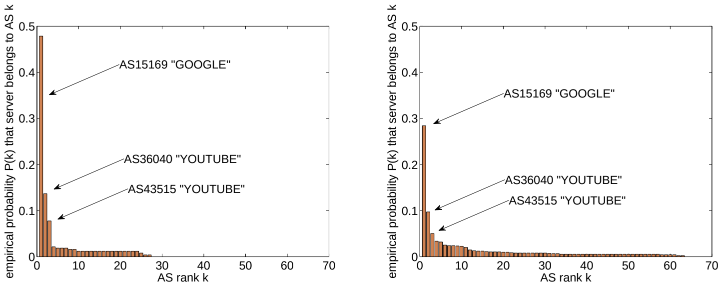
iments in distributed network measurements. Although both sets of measurement points are globally distributed the fraction of the CDN which is discovered by the probes has very different characteristics.

The amount of servers which is located in the US almost doubles for the PlanetLab measurement. While 44% of the requests are resolved to US servers in the Crowdsourcing measurement, nearly all requests of PlanetLab nodes are served by YouTube servers located in the US. Although less than 15% of clients are in US, requests are frequently directed to servers in the US. That means that there is still potential to further distribute the content in the CDN.

Coverage of Autonomous Systems with YouTube Servers

To identify the distribution of clients on ISPs and to investigate the expansion of CDNs on autonomous systems we map the measurement points to the corresponding autonomous systems.

Figure 1.12a shows the autonomous systems of YouTube servers accessed by PlanetLab nodes. The autonomous systems were ranked by the number of YouTube servers located in the AS. The empirical probability $P(k)$ that a server



(a) PlanetLab

(b) Crowdsourcing

Figure 1.12: Distribution of YouTube servers on autonomous systems from a) PlanetLab and b) Crowdsourcing perspective.

belongs to AS with rank k is depicted against the AS rank. The number of autonomous systems hosting YouTube servers that are accessed by PlanetLab nodes is limited to less than 30. The top three ranked ASes are AS15169, AS36040 and AS43515. AS15169 is the Google autonomous system which includes the Google backbone. The Google backbone is a global network that reaches to worldwide points of presence to offer peering agreements at peering points. AS36040 is the YouTube network connecting the main datacenter in Mountain-View which is also managed by Google. AS43515 belongs to the YouTube site in Europe which is administrated in Ireland. Hence, two thirds of the servers are located in an autonomous systems which is managed by Google. Only few requests are served from datacenters not being located in a Google AS. The reason that request from PlanetLab are most frequently served by ASes owned by Google might be a good interconnection of the NRENs to the Google ASes.

Figure 1.12b depicts the autonomous systems where requests to YouTube videos from the crowdsourcing workers were directed. The empirical probability that a server belongs to an AS has been plotted dependent on the AS rank. The YouTube servers identified by the crowdsourcing probes are located in more than 60 autonomous systems. Hence, the YouTube CDN is expanded on a higher range of ASes from the crowdsourcing perspective compared to PlanetLab. Again the three autonomous systems serving most requests are the ASes managed by Google, respectively YouTube. But the total number of requests served by a Google managed AS is only 41%. Hence, in contrary to the PlanetLab measurement, requests are served most frequently from ASes not owned by Google. Here, caches at local ISPs managed by YouTube could be used to bring the content close to users without providing own infrastructure. This would also explain the large number of identified ASes providing a YouTube server. The results show that the PlanetLab platform is not capable to measure the structure of a global CDN, since large parts of the CDN are not accessed by clients in NRENs.

1.4 Distribution of IP Addresses on Autonomous Systems

The performance of systems using CPE or resources provided by end-users depend on the capacity and number of devices available. To assess the potential of a hierarchical cache system in an ISPs network, the number of active subscribers in an autonomous systems has to be known. Assuming that the number of active IP-addresses is correlated to the number of subscribers in an autonomous system, we use the Internet Census Dataset to determine the distribution of active IP-addresses on autonomous systems.

The Internet Census Dataset[60] was conducted from June to October 2012. The complete IPv4-address room was scanned using a bot-net consisting of 4,200,000 nodes. In the ICMP ping scan more than 420 million replied to requests more than once. The service probe data reveal open ports on devices which is used to infer the type of device. The Internet Census Dataset was validated forensically in [46]. In [47] the scope of the dataset is taken into perspective and show that, although there are some qualitative problems, the measurement data seems to be authentic.

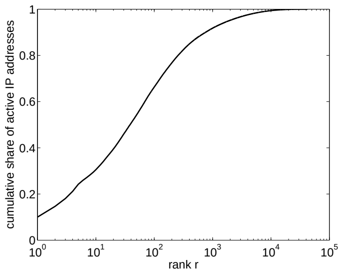


Figure 1.13: Cumulative share of active IP-addresses in autonmous systems ranked in descending order.

We use an IP to ASN mapping to derive the autonomous system number for each IP-address. There are different services, that provide an IP to ASN mapping. The whois-service can be used to get the current ASN for an IP-address. To enable an efficient evaluation we used the MAXMIND GeoLite ASN database [61], which is updated every month and can be downloaded and used as a local database. The results of the MAXMIND GeoLite ASN database were cross checked with results obtained from whois, which showed no differences.

The ICMP ping scan discovered a total of 598,180,914 IP-addresses. The service probe scan discovered 244,000 IP-addresses that listen to port 9100 and are identified as print servers, and 70.84 million IP-addresses of web-servers that listen to port 80. Assuming that most network functions do not reply to ICMP ping requests and neglecting different network functions, this results in 88.1% of IP-addresses assigned to end-user devices. Since the Internet Census the number of Internet users increased, which also has to be considered. According to [62] there is a 7% annual increase in fixed-broadband subscriptions in the past three years.

Figure 1.13 shows the cumulative share of active IP-addresses in the autonomous systems ranked in descending order. The 100 largest autonomous systems make up 2/3 of active IPs and more than 85% of the IPs are active in only 1%

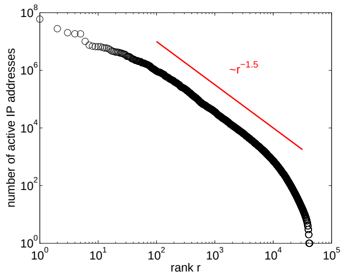


Figure 1.14: Rank of Internet providers with number of active IP-addresses per AS.

Table 1.3: Rank of top 5 provider with most active IP-addresses.

rank r	ASN	provider	# active IPs
1	4134	ChinaTelecom	59,824,824
2	4837	China-Network-Communication-Group	27,776,643
3	7922	Comcast	20,227,918
4	4766	KoreaTelecom	18,502,963
5	3320	DeutscheTelekomAG	18,476,519

of the autonomous systems. The 10 largest autonomous systems already contain 30% of the active IPs.

Figure 1.14 shows the number of active IP-addresses per AS ranked in descending order. The top 5 ASes are shown in table 1.3. The AS with most active IP-addresses is ChinaTelecom with almost 60 million active IPs, followed by another Chinese provider. The largest AS in the US is Comcast on rank three. The largest Korean and German providers are ranked 4 and 5 with more than 18 million active IPs. The number of active IP addresses can be approximated with a power law with slope 1.5 that drops a little for low ranks. This shows that the distribution of active IP addresses on ASes is highly heterogeneous. That means the potential of approaches leveraging spare resources on home gateways depends on the AS.

1.5 Lessons Learned

In this chapter we characterized content delivery networks on autonomous systems level. For that purpose we used measurements conducted on the distributed platform PlanetLab and a crowdsourcing platform. To assess the potential of peer assisted content delivery approaches, we determined the number of active IP-addresses from the Internet Census dataset.

First, we have investigated where in the Internet BitTorrent traffic is located and which ISPs benefit from its optimization. To this end, we used measurements of live BitTorrent swarms to derive the location of BitTorrent peers and data provided by Caida.org in order to calculate the actual AS path between any two peers. Our results show that the traffic optimization potential depends heavily on the type of ISP. Different ISPs will pursue different strategies to increase revenues. Our results confirm that selecting peers based on their locality has a high potential to shorten AS paths between peers and to optimize the overlay network. In the observed BitTorrent swarms twice as much traffic can be kept intra-AS using locality peer selection. Thus, the inter-AS traffic is almost reduced by 50 % in tier−1 and in large ISPs.

Second, we proposed the usage of crowdsourcing platforms for distributed network measurements to increase the coverage of vantage points. We evaluated the capability to discover global networks by comparing the coverage of video server detected using a crowdsourcing platform as opposed to using the PlanetLab platform. To this end, we used exemplary measurements of the global video CDN YouTube, conducted in both the PlanetLab platform as well as the crowdsourcing platform Microworkers. Our results show that the vantage points of the concurring measurement platforms have very different characteristics. We could show that the distribution of vantage points has high impact on the capability of measuring a global content distribution network. The capability of PlanetLab to measure a global CDNs is rather low, since 80% of requests are directed to the United States. Our results confirm that the coverage of vantage points is increased by crowdsourcing. Using the crowdsourcing platform we obtain a diverse set of vantage points that reveals more than twice as many autonomous systems deploying video servers than the widely used PlanetLab platform.

Finally, we analyzed the Internet census dataset to derive the distribution of IP-addresses on autonomous systems. To this end, we used a mapping of IP-addresses to autonomous system numbers. we find that the distribution of IP-addresses is highly heterogeneous showing that 30% of the active IPs belong to the 10 largest autonomous systems. This means that the potential of approaches that use resources of home gateways highly depend on the ISP network.

Based on the results obtained in this chapter, we develop models that describe the characteristics of CDNs and the number of active subscribers in ISP networks. The models allow us to analyze the performance of traffic management mechanisms in realistic scenarios.

Bibliography and References

Bibliography of the Author

General References

[1] G. Petropoulos, A. Lareida, V. Burger, M. Seufert, S. Soursos, and B. Stiller, "Wifi offloading and socially aware prefetching on augmented home routers", in *40th Conference on Local Computer Networks (LCN) 2015*, Clearwater Beach, FL, USA: IEEE, Oct. 2015, pp. 1–3. [Online]. Available: https://files.ifi.uzh.ch/CSG/staff/lareida/Extern/Publications/rbhorst-demo.pdf.

[2] V. Burger, G. Darzanos, I. Papafili, and M. Seufert, "Trade-off between qoe and operational cost in edge resource supported video streaming", in *10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC)*, Krakow, Poland, Nov. 2015.

[3] V. Burger, J. Frances Pajo, O. R. Sanchez, M. Seufert, C. Schwartz, F. Wamser, F. Davoli, and P. Tran-Gia, "Load dynamics of a multiplayer online battle area and simulative assessment of edge server placements", in *ACM Multimedia Systems Conference (MMSys)*, Klagenfurt, Austria, May 2016.

[4] V. Burger, D. Hock, I. Scholtes, T. Hoßfeld, D. Garcia, and M. Seufert, "Social network analysis in the enterprise: Challenges and opportunities", in *Socioinformatics - The Social Impact of Interactions between Humans and

IT, K. Zweig, W. Neuser, V. Pipek, M. Rohde, and I. Scholtes, Eds., Springer International Publishing, Aug. 2014.

[5] V. Burger, T. Hoßfeld, D. Garcia, M. Seufert, I. Scholtes, and D. Hock, "Resilience in enterprise social networks", in *Workshop Sozioinformatik 2013*, Koblenz, Germany, Sep. 2013.

[6] V. Burger, F. Kaup, M. Seufert, M. Wichtlhuber, D. Hausheer, and P. Tran-Gia, "Energy considerations for wifi offloading of video streaming", in *7th EAI International Conference on Mobile Networks and Management (MONAMI)*, Santander, Spain, Sep. 2015.

[7] V. Burger, M. Seufert, T. Hoßfeld, and P. Tran-Gia, "Performance evaluation of backhaul bandwidth aggregation using a partial sharing scheme", *Physical Communication*, vol. 19, pp. 135–144, Jun. 2016.

[8] V. Burger, M. Seufert, F. Kaup, M. Wichtlhuber, D. Hausheer, and P. Tran-Gia, "Impact of wifi offloading on video streaming qoe in urban environments", in *IEEE Workshop on Quality of Experience-based Management for Future Internet Applications and Services (QoE-FI)*, London, UK, Jun. 2015.

[9] T. Hoßfeld, V. Burger, H. Hinrichsen, M. Hirth, and P. Tran-Gia, "On the computation of entropy production in stationary social networks", *Social Network Analysis and Mining*, vol. 4, Apr. 2014.

[10] A. Lareida, G. Petropoulos, V. Burger, M. Seufert, S. Soursos, and B. Stiller, "Augmenting home routers for socially-aware traffic management", in *40th Annual IEEE Conference on Local Computer Networks (LCN)*, Clearwater Beach, FL, USA, Oct. 2015.

[11] D. Schlosser, M. Jarschel, V. Burger, and R. Pries, "Monitoring the user perceived quality of silk-based voice calls", in *Australasian Telecommunication Networks and Applications Conference (ATNAC 2010)*, Auckland, New Zealand, Nov. 2010.

[12] M. Seufert, V. Burger, and T. Hoßfeld, "Horst - home router sharing based on trust", in *Workshop on Social-aware Economic Traffic Management for Overlay and Cloud Applications (SETM)*, Zurich, Switzerland, Oct. 2013.

[13] M. Seufert, V. Burger, K. Lorey, A. Seith, F. Loh, and P. Tran-Gia, "Assessment of subjective influence and trust with an online social network game", *Computers in Human Behavior*, 2016.

[14] M. Seufert, V. Burger, F. Wamser, P. Tran-Gia, C. Moldovan, and T. Hoßfeld, "Utilizing home router caches to augment cdns towards information-centric networking", in *European Conference on Networks and Communications (EuCNC)*, Paris, France, Jun. 2015.

[15] M. Seufert, G. Darzanos, V. Burger, I. Papafili, and T. Hoßfeld, "Socially-aware traffic management", in *Workshop Sozioinformatik*, Koblenz, Germany, Sep. 2013.

[16] M. Seufert, G. Darzanos, I. Papafili, R. Łapacz, V. Burger, and T. Hoßfeld, "Socially-aware traffic management", in *Socioinformatics - The Social Impact of Interactions between Humans and IT*, K. Zweig, W. Neuser, V. Pipek, M. Rohde, and I. Scholtes, Eds., Springer International Publishing, Aug. 2014.

[17] M. Seufert, T. Griepentrog, and V. Burger, "A simple wifi hotspot model for cities", *IEEE Communications Letters*, vol. 20, no. 2, pp. 384–387, Feb. 2016.

[18] M. Seufert, T. Hoßfeld, A. Schwind, V. Burger, and P. Tran-Gia, "Group-based communication in whatsapp", in *1st IFIP Internet of People Workshop (IoP)*, Vienna, Austria, May 2016.

[19] Cisco, "Forecast and methodology, 2015–2020", *Cisco Visual Networking Index*, 2016.

[20] V. Valancius, N. Laoutaris, L. Massoulié, C. Diot, and P. Rodriguez, “Greening the internet with nano data centers”, in *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, ACM, 2009, pp. 37–48.

[21] V. Burger, F. Lehrieder, T. Hoßfeld, and J. Seedorf, “Who profits from peer-to-peer file-sharing? traffic optimization potential in bittorrent swarms”, in *International Teletraffic Congress (ITC 24)*, Krakow, Poland, Sep. 2012.

[22] V. Burger, M. Hirth, C. Schwartz, and T. Hoßfeld, “Increasing the coverage of vantage points in distributed active network measurements by crowdsourcing”, in *Measurement, Modelling and Evaluation of Computing Systems (MMB 2014)*, Bamberg, Germany, Mar. 2014.

[23] V. Burger and T. Zinner, “Performance analysis of hierarchical caching systems with bandwidth constraints”, *ITNAC*, 2016.

[24] B. Cohen, “Incentives build robustness in BitTorrent”, in *1st Workshop on the Economics of Peer–to-Peer Systems*, 2003.

[25] T. Hoßfeld, F. Lehrieder, D. Hock, S. Oechsner, Z. Despotovic, W. Kellerer, and M. Michel, “Characterization of BitTorrent swarms and their distribution in the Internet”, *Computer Networks*, vol. 55, no. 5, pp. 1197–1215, 2011. DOI: `10.1016/j.comnet.2010.11.011`.

[26] M. Kryczka, R. Cuevas, C. Guerrero, and A. Azcorra, “Unrevealing the structure of live BitTorrent swarms: Methodology and analysis”, in *Proc. IEEE P2P*, 2011.

[27] M. Izal, G. Urvoy-Keller, E. W. Biersack, P. A. Felber, A. A. Hamra, and L. Garces-Erice, “Dissecting BitTorrent: Five months in a torrent’s lifetime”, in *Proc. PAM*, 2004.

[28] J. Pouwelse, P. Garbacki, D. Epema, and H. Sips, “The Bittorrent P2P file-sharing system: Measurements and analysis”, in *Proc. IPTPS*, 2005.

[29] L. Guo, S. Chen, Z. Xiao, E. Tan, X. Ding, and X. Zhang, "Measurements, analysis, and modeling of BitTorrent-like systems", in *Proc. SIGCOMM IMC*, 2005, pp. 4–4.

[30] C. Zhang, P. Dhungel, D. Wu, and K. W. Ross, "Unraveling the BitTorrent ecosystem", *IEEE Transactions on Parallel and Distributed Systems*, vol. 99, 2010, ISSN: 1045-9219. DOI: `http://doi.ieeecomputersociety.org/10.1109/TPDS.2010.123`.

[31] P. Tran-Gia, T. Hoßfeld, M. Menth, and R. Pries, "Emerging issues in current future internet design", *E&i Elektrotechnik und Informationstechnik, Special Issue 'Future Internet'*, vol. 07/08, 2009.

[32] L. Gao, "On inferring autonomous system relationships in the Internet", *IEEE/ACM Trans. Netw.*, vol. 9, no. 6, pp. 733–745, 2001.

[33] X. Dimitropoulos, D. Krioukov, M. Fomenkov, B. Huffaker, Y. Hyun, G. Riley, *et al.*, "AS relationships: Inference and validation", *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 1, pp. 29–40, 2007.

[34] G. Di Battista, M. Patrignani, and M. Pizzonia, "Computing the types of the relationships between autonomous systems", in *Proc. INFOCOM*, vol. 1, 2003, pp. 156–165.

[35] Google, *Peering & content delivery*, https://peering.google.com/.

[36] V. Adhikari, S. Jain, Y. Chen, and Z. Zhang, "Vivisecting youtube: an active measurement study", in *Proceedings IEEE INFOCOM*, 2012.

[37] P. Tran-Gia, T. Hoßfeld, M. Hartmann, and M. Hirth, "Crowdsourcing and its impact on future internet usage", *It - Information Technology*, vol. 55, 2013.

[38] L. Von Ahn, B. Maurer, C. McMillen, D. Abraham, and M. Blum, "Recaptcha: Human-based character recognition via web security measures", *Science*, no. 5895, 2008.

[39] InnoCentive, Inc, *Innocentive*, http://www.innocentive.com/.

[40] T. Hoßfeld, M. Hirth, and P. Tran-Gia, “Modeling of crowdsourcing platforms and granularity of work organization in future internet”, in *In Proceedings of the International Teletraffic Congress (ITC)*, 2011.

[41] R. Torres, A. Finamore, J. R. Kim, M. Mellia, M. M. Munafo, and S. Rao, “Dissecting video server selection strategies in the youtube cdn”, in *31st International Conference on Distributed Computing Systems (ICDCS)*, 2011.

[42] V. Adhikari, S. Jain, and Z. Zhang, “Where do you "tube"? uncovering youtube server selection strategy”, in *IEEE ICCCN*, 2011.

[43] A. Rafetseder, F. Metzger, D. Stezenbach, and K. Tutschku, “Exploring youtube’s content distribution network through distributed application-layer measurements: A first view”, in *Proceedings of the 2011 International Workshop on Modeling, Analysis, and Control of Complex Networks*, 2011.

[44] Z. S. Bischof, J. S. Otto, M. A. Sánchez, J. P. Rula, D. R. Choffnes, and F. E. Bustamante, “Crowdsourcing isp characterization to the network edge”, in *Proceedings of the first ACM SIGCOMM workshop on Measurements up the stack*, 2011.

[45] T. Hoßfeld, R. Schatz, E. Biersack, and L. Plissonneau, “Internet video delivery in youtube: from traffic measurements to quality of experience”, in *Data Traffic Monitoring and Analysis: From measurement, classification and anomaly detection to Quality of experience*, M. M. Ernst Biersack Christian Callegari, Ed., Springer’s Computer Communications and Networks series, Volume 7754, 2013.

[46] A. Dainotti and A. King, *Caida blog: Carna botnet scans confirmed.*

[47] T. Krenc, O. Hohlfeld, and A. Feldmann, “An internet census taken by an illegal botnet: A qualitative assessment of published measurements”, *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 3, pp. 103–111, 2014.

[48] F. Wamser, R. Pries, D. Staehle, K. Heck, and P. Tran-Gia, "Traffic characterization of a residential wireless Internet access", *Special Issue of the Telecommunication Systems (TS) Journal*, vol. 48: 1-2, 2010.

[49] G. Yang and W. Dou, "An efficient algorithm for AS path inferring", in *Proc. ICHIT*, 2009, pp. 151–154.

[50] *The CAIDA AS relationships dataset, as-rel.2011.01.16.txt*. http://www.caida.org/data/active/as-relationships/.

[51] *Internet topology collection*, http://irl.cs.ucla.edu/topology/.

[52] H. Xie, Y. R. Yang, A. Krishnamurthy, Y. G. Liu, and A. Silberschatz, "P4p: Provider portal for application", *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 4, pp. 351–362, 2008.

[53] V. Aggarwal, A. Feldmann, and C. Scheideler, "Can ISPs and P2P users cooperate for improved performance?", *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 3, pp. 29–40, 2007.

[54] T. Hoßfeld et al., "An economic traffic management approach to enable the triplewin for users, ISPs, and overlay providers", in *FIA Prague Book, ISBN 978-1-60750-007-0*, 2009.

[55] Cisco, "Forecast and methodology, 2012–2017", *Cisco Visual Networking Index*, 2013.

[56] C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian, "Internet inter-domain traffic", in *ACM SIGCOMM Computer Communication Review*, 2010.

[57] PlanetLab, *An open platform for developing, deploying, and accessing planetary-scale services*, 2009. [Online]. Available: http://www.planet-lab.org/.

[58] Microworkers, *Work & earn or offer a micro job*, http://microworkers.com/.

[59] MaxMind, *Geolite databases*, http://dev.maxmind.com/geoip/geolite/.

[] M. Hirth, T. Hoßfeld, and P. Tran-Gia, "Anatomy of a crowdsourcing platform - using the example of microworkers.com", in *Workshop on Future Internet and Next Generation Networks (FINGNet)*, Seoul, Korea, 2011.

[60] B. Carna, *Internet Census 2012 Port Scanning /0 Using Insecure Embedded Devices*, 2013. [Online]. Available: `http://internetcensus2012.bitbucket.org/paper.html`.

[61] MaxMind, *Geolocation service*. [Online]. Available: `http://www.maxmind.com/`.

[62] International Telecommunication Union, "Ict facts and figures – the world in 2015", *Report*, 2015.

ISSN 1432-8801