INTERNATIONAL JOURNAL OF COMMUNICATION SYSTEMS
Int. J. Commun. Syst. 2006; **19**:95–114
Published online 14 November 2005 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/dac.750

On properties of multicast routing trees

Milena Janic[‡] and Piet Van Mieghem[*,†]

Delft University of Technology, Faculty of EEMCS, 2600 GA Delft, The Netherlands

SUMMARY

In the last several years we witnessed the proliferation of multimedia applications on the Internet. One of the unavoidable techniques to support this type of communication is multicasting. However, even a decade after its initial proposal, multicast is still not widely deployed. One of the reasons is the lack of a solid business model. If the gain and the cost of multicast could be predicted, network operators might be encouraged to deploy multicast on a larger scale. In this paper we propose analytical expressions that could be used to estimate the gain of network-layer multicast. We show that the theoretical model matches extensive simulation and Internet measurement results remarkably well.

Furthermore, we examine the reliability of *traceroute* data and of *traceroutes*-based conclusions. We investigate the node degree distributions in the Internet maps obtained from CAIDA and RIPE and we show the divergency of our results with those obtained by other researchers. We further focus on the analysis of multicast trees based on *traceroute* data. Only few results have been available on the node degree distribution of multicast routing trees which provided contradictory conclusions. Our results seem to indicate that the node degrees follow power laws only for a large number of multicast users. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: multicast; shortest path trees; node degree distribution; traceroutes

1. INTRODUCTION

The number of multimedia applications on the Internet, combining audio, video and data streams, is growing explosively. Multimedia applications, even when data compression is used, require in general a considerable amount of bandwidth, and they are often delay sensitive. Such applications include radio/television broadcast, desktop video/audio conferencing, shared white boards, tele-classing, file transfers to multiple locations, online gaming and animated simulations. IP multicast, offering a scalable point-to-multipoint delivery, is regarded as a promising network service for group multimedia applications.

Even though the first deployment of multicast occurred in 1992, and in spite of the continuously rising demand for a ubiquitous multicast service, IP multicast is still experiencing

[*] Correspondence to: Piet Van Mieghem, Delft University of Technology, Faculty of EEMCS, 2600 GA Delft, The Netherlands.
[†] E-mail: P.VanMieghem@ewi.tudelft.nl
[‡] E-mail: M.Janic@ewi.tudelft.nl

Received 1 September 2004
Revised 1 March 2005
Accepted 1 May 2005

slow wide-scale deployment. One of the reasons is the lack of a proper business model. The computational and administrational overhead of multicast group management increases the deployment cost compared to the cost of unicast. Clearly, the deployment of multicast can only be justified if the nett gain defined as savings minus costs is larger than the nett gain for unicast. We believe that understanding the multicast tree characteristics and establishing a good realistic model for multicast could accelerate the deployment of multicast on a larger scale.

Recent studies on topological properties of the Internet at the router as well as the autonomous system (AS) level have attracted considerable attention. Passive and active measurements are used to get more insight into the fundamental topological properties of Internet. Mostly the *traceroute* utility [1] has been used for acquiring a map of the Internet. One of the most striking observations in the Internet graphs is the long-tailed node degree distribution. However, as we will see further in the paper, care is needed when extrapolating the results obtained via *traceroute* measurements to the Internet as a whole. We will also show that, based on *traceroute* data, the node degree distribution in the Internet map does not always follow a power law.

Oddly, whereas the structure of the Internet topology has been the focus of many researchers, modelling multicast trees has not received the attention it deserves. Similar to the node degree distribution in the Internet map, we will show that the node degree distribution in the multicast tree does not always follow a power law.

In addition to the node degree distribution, we have investigated the gain and the cost of multicast trees. One possible criterion to assess the gain of multicast trees is the number of hops or links used in the tree rooted at a particular source to m randomly chosen destinations. Alternatively, the cost of multicast trees can be defined as the sum of all the link weights in a tree connecting m uniformly chosen nodes. We have analysed the gain and cost in two approaches to multicast realization. The minimum cost approach is to construct a single tree to distribute the traffic from all senders in the group, regardless of the sender's location, and to minimize the total weight of the tree. Hence, it optimizes the use of network resources. The problem of finding a minimum weight tree that spans all multicast users is known as the Steiner tree problem [2]. However, the computational complexity of finding a Steiner minimum tree (SMT), proven to be *NP-complete*, together with its less stable dynamic behaviour [3] prohibits the implementation of this algorithm for multicast routing protocols on Internet. Instead, most of the current Internet protocols forward packets based on the (reverse) shortest path. A shortest path tree (SPT) is a union of the shortest paths from the source node to the destinations. The SPT algorithm does not necessarily result in a tree that economizes on network resources but it is easy to compute and it offers a minimum delay. Moreover, in Reference [4] van der Hofstad *et al.* have shown that in the complete graph with N nodes and with i.i.d. exponential link weights with mean 1, the average weight in the SPT is smaller than $\zeta(2)$, where $\zeta(z)$ is Riemann Zeta function. Since the average weight of the SMT in the same graph is limited by $\zeta(3)$, the ratio $[\zeta(3)/\zeta(2)] - 1 = 0.37$ implies that on average the performance of the SPT is not more than 37% worse than that of SMT. In this paper we propose analytic expressions that can be used to compute the gain of the SPT, which we also compare with the SMT and with Internet measurement data.

The remainder of this paper is organized as follows: In Section 2 the collection of measurement data is explained. In Section 3 we briefly discuss the ambiguities in *traceroute* data and the dangers of drawing conclusions based on these data. In Section 3.4, we focus on a problem specific for multicast analysis of unicast *traceroute* measurements: the occurrence of cycles. A detailed study of the node degree distributions in the Internet map is presented in

Section 4. The comparison of measurements with theory and simulations is discussed in Section 5. Finally, we conclude in Section 6.

2. MEASUREMENT DATA SETS

2.1. RIPE-NCC

In our analysis we have used *traceroute* data provided by RIPE NCC[§] (the Network Coordination Centre of the Réseaux IP Européen). RIPE NCC performs *traceroute* measurements between measurement boxes scattered over Europe (and few in the U.S. and New Zealand). At the time of writing, the number of boxes has been 92 with on average 2 boxes being added per month. The *traceroute* data has been collected in two periods: 1st May–1st June 2003 and 1st January–1st February 2004. Since not all the boxes are active all the time, the number of boxes from which we obtained the data is smaller than the total number of boxes. In the period 1st May–1st June 2003, from each of the 60 boxes, *traceroutes* to all the other test boxes have been obtained, resulting in a total of 1 329 019 *traceroute*s. Among all the distinguished paths in the database, only the most dominant one (the one being returned by *traceroute* most frequently) has been considered. After discarding the erroneous data, 973 non-erroneous most dominant paths were distinguished (data set 1). We have performed the same measurements in the period 1st January–1st February 2004, and collected *traceroute* paths from 72 sources to a variable number of destinations (60–70), representing the multicast users. The number of collected dominant non-erroneous *traceroute*s paths in this experiment was 4521 (data set 2). Since PIM-SM (the most commonly used multicast routing protocol) relies on unicast routing tables for routing, trees constructed as union of *traceroute*s will resemble multicast routing trees.

2.2. CAIDA

CAIDA's Skitter tool[¶] deploys a method similar to *traceroute* to determine the IP path to a destination. Destinations are chosen from BGP tables and a database of Web servers. Skitter sends ICMP echo request packets, increments the TTL when sending them and registers the IP address of the replying routers. If a router does not respond to three subsequent ICMP request packets, the TTL is increased. When the desired destination is reached, Skitter registers the round-trip-time (RTT) as well. However, if the TTL equals 30, 'ICMP unreachable' reply are received, or if a routing loop is encountered, then Skitter stops probing the destination. Resolving the aliasing problem is attempted as well, by deploying the *iffinder* tool. The *iffinder* tool relies on a similar router identification technique as the one described in Reference [5].

The CAIDA Skitter project has deployed around 30 monitors worldwide. Each monitor performs *traceroutes* measurements to thousands of destinations every day. We have obtained the *traceroutes* from 6 skitter monitors over 3 days measurements (1, 2, 3 April 2003). Totally 684 135 *traceroute*s have been collected in our database. The incomplete traces have been eliminated, leaving 276 680 out of 684 135 complete and stable *traceroutes* in the database.

[§] http://www.ripe.net

[¶] http://www.caida.org

Multicast routing trees have been obtained in the following way: first, we have randomly chosen $m = 50, 100, 500, 1000$ destinations (multicast users). For three monitor boxes (two of them situated in United States and one in Japan) the collection of paths from these three sources to randomly chosen destinations has been obtained via *traceroute*. In this way, we obtained for each source a set of 4 trees. These trees resemble multicast routing trees, under the assumption that multicast group members are uniformly distributed.

2.3. *PlanetLab*

PlanetLab[||] is an open, worldwide distributed testbed that enables performing experiments under real-world conditions, and on a large scale. At the time of writing, there were more than 200 institutions participating in PlanetLab projects, including TUDelft. Our experiments have been executed on 10th November 2004. At that moment, there were 445 PlanetLab nodes running on locations in U.S.A., Asia and Europe. Architecturally, the PlanetLab network is similar to RIPE: each node can serve as a source as well as a destination. From each of the PlanetLab sites, we can perform *traceroutes* to all the other PlanetLab sites. Since in some cases there are multiple nodes per PlanetLab site (situated at the same location), we selected one node per PlanetLab site, resulting in totally 79 nodes (since many nodes are not active or not accessible at all time).

3. PROBLEMS WITH TRACEROUTES

The *traceroute* utility [1] has been the most popular tool for acquiring a map of Internet so far. *Traceroute* infers an IP path between a source and a destination by sending out probe packets with progressively increasing TTLs and then analysing the ICMP error responses sent by routers along the path receiving a packet with a zero TTL. The Internet map can be created as a union of these *traceroute* paths. Nevertheless, there are many issues considering *traceroute* measurements that complicate this apparently straightforward procedure.

3.1. *Inaccuracies*

In spite of being the best current tool for inferring the end-to-end IP-level path, *traceroutes* suffer from several types of errors and flaws [6]. Previously [7], we have ascertained that 17% of the collected *traceroutes* were erroneous (loops, etc.). In addition, since two probes are sent to every router on the path, a considerable amount of overhead is generated. Furthermore, some ISPs hide their routers from *traceroute* by manipulating the ICMP replies. This can reduce the accuracy of topologies discovered. Finally, as reported in Reference [8], when performing *traceroute* measurements using different tools (Skitter and Rocketfuel) in the same area of interest (with the time difference of two months) a noticeable number of different routers and links have been found.

[||] http://www.planet-lab.org

3.2. *Alias resolution*

The *traceroute* utility returns the list of IP addresses of routers along the path from source to destination. One router can have several interfaces, with several different IP addresses. In order to obtain accurate router-level maps, it is necessary to determine which IP addresses belong to the same router. This, however, is not a trivial problem. The early Internet mapping attempts have either ignored alias resolution [9], or have used a very simple alias probe heuristic [10]: after sending an alias probe packet to a non-existing port, the router responds with an ICMP *port unreachable* message. If the source address of this packet is different from the address to which it has been sent, then these two addresses represent two interfaces belonging to the same router. The second in the row is alias resolution heuristic proposed by Govindan and Tangmunarunkit in Mercator [5], that relies on alias probing, but includes two refinements. The most enhanced alias resolving technique up to now, has been utilized in Rocketfuel [8]. Rocketfuel combines Mercator's address-based heuristic with other techniques, the most important of which is comparing the IP identifier fields. The IP identifier-based method significantly outperforms the Mercator heuristic: it finds almost three times as many aliases as an address-based method. Nevertheless, in spite of being the most effective alias resolving technique, Rocketfuel's Ally tool still encountered some substantial problems, such as unresponsive IP addresses (almost 6000 out of 56 000 addresses did not react when queried for aliases).

3.3. *Bias sampling*

Lakhina *et al.* [11] have pointed to the bias in sampling when deducing topological properties of maps based on *traceroute* measurements. Currently, most of the *traceroute* measurements are performed from a limited number of publicly available sources, to a larger number of more flexibly chosen destinations. Consequently, nodes and links lying nearer to the sources will be visited much more frequently than those that are more remote, forming a possible cause for the manifestation of the long-tailed node degree distribution.

The same authors raise another significant problem: if a graph is generated by aggregating shortest paths from a limited number of sources and destinations, the node degree distribution in those graphs can vary significantly. They demonstrated via simulations that the Erdös-Rényi random graphs $G_p(N)$ [12] that possess a binomial node degree distribution, will appear to have a power-law characteristic. This implies that it is fallacious to characterize the router-level topology of Internet based on the node degree distribution of its (imperfect) subgraph. For these reasons, we avoid to extrapolate conclusions obtained from a particular *traceroute* data set to the whole Internet router-level topology.

3.4. *Analysis of cycles*

The topology of a multicast tree is one of the significant elements for successful multicast management. Attaining this data is, however, a complex task, much more difficult than tracing unicast paths. The determination of the multicast tree requires accessing the data in the routers themselves. We have assumed that paths constituting a multicast tree are actually the ones returned by *traceroute* measurements. The MBONE era, consisted of tunnelling and DVMRP, is long behind us. An increasing number of core network operators implement native multicast today. Since PIM-SM relies on unicast routing tables, the assumption we make seems justified (Table I).

However, the topology of the union of *traceroutes* can include cycles, and therefore is not a tree. When tracing paths from source X to m different destinations, the subsections of paths between the same two nodes may consist of different nodes as illustrated in Figure 1. In Figure 1(a), for three destinations that have been traced from a source 12, a path from the source 12 has traversed nodes 10 and 665. In two out of three cases *traceroute* has returned nodes 196 and 666 as intermediate nodes between 10 and 665, whereas in the *traceroute* record for the third destination, destination nodes 197 and 195 have been detected. In the real multicast session, the corresponding multicast tree would consist either of segment 10-196-666-665 or 10-197-195-665, but not of both of them. In our example, since the majority of paths (two out of three) have traversed the nodes 10-196-666-665, we assumed that this segment would appear in the multicast tree. Actually, the choice of either path in Figure 1(a) does not influence neither the node degree distribution nor the total number of links in the tree. We call this type of cycles *symmetrical*. Not all the loops detected in our data are *symmetrical*. The examples depicted in Figure 1(b) and (c) show *asymmetrical* loops. The cycle in Figure 1(c) is seemingly *symmetrical*, however, the existence of children of the node 293 (an intermediate node in one segment) affects the total number of links and the degree distribution as well.

In the RIPE-NCC *traceroute* data set 2, a total of 606 loops in 72 created trees has been detected, out of which 493 (81%) symmetrical. The loops occur probably due to load balancing and router/link failures. The distribution of the number of loops per source node is shown in Figure 2(a).

The hopcount distribution in the loops shown in Figure 2(b) suggests that $\Pr[\text{hop} = k] \sim e^{-\alpha k}$, with $0.7 < \alpha < 1$, which implies that most loops are short.

Table I. The average number of loops and symmetrical loops per tree (source).

Data set	# traces	# srcs	# dests	# loops-aver.	# sym. loops-aver.
RIPE	4521	72	60–70	7.66	6.75

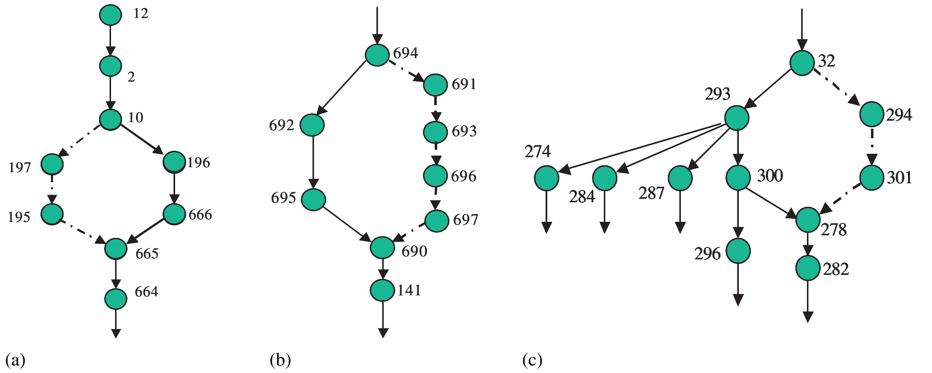

Figure 1. The illustration of cycles: (a) symmetrical; (b) asymmetrical type 1; and (c) asymmetrical type 2.

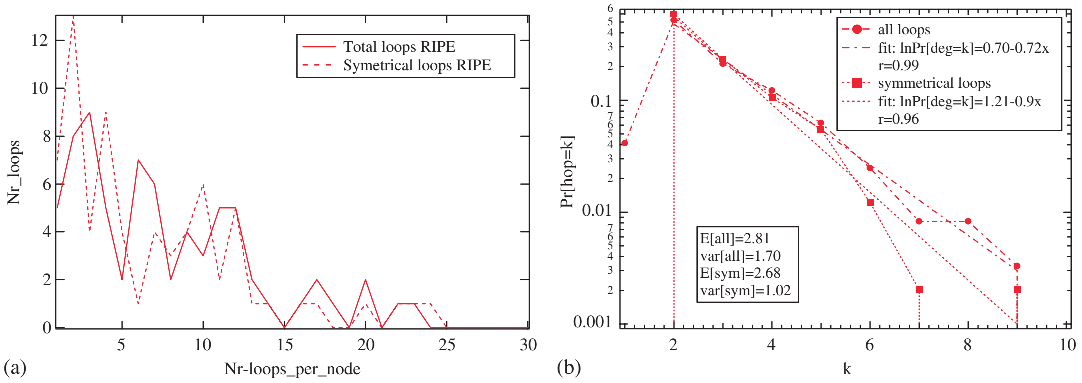

Figure 2. (a) The number of loops per source node; and (b) the distribution of the hopcount in the loops.

4. NODE DEGREE DISTRIBUTIONS IN INTERNET

4.1. Related work

One of the first attempts to map the Internet router-level topology was that of Pansiot and Grad [10]. Pansiot and Grad have constructed the Internet map based upon the *traceroute* records from a single node to 5000 geographically dispersed destinations, as well as on *traceroutes* from a subset of 11 nodes chosen from the set of 5000 nodes to the rest of the destinations. Upon these records, they created a graph containing 3888 nodes and 4857 edges. Although they have performed a simple heuristic for resolving aliases, i.e. determining which IP addresses belong to the same router, and have discovered in that way 200 aliases (5% of the total number of nodes), some apparently different nodes actually represent the same node. The degree distribution of the nodes in the aforementioned graph is presented in Figure 3, based on the data presented in Reference [10]. Subsequent to Pansiot and Grad's attempt, several global router-level Internet mapping projects have been initiated, almost all based on the *traceroute* utility [5, 9].

Burch and Cheswick [9] have used BGP backbone routing tables in order to determine the destinations of traceroutes. For each prefix in the table, they repeatedly generated a randomly chosen IP address from within that prefix. From traceroutes to each such address, they determine router adjacencies, building a router-level map in this manner, without applying any alias resolving technique.

Govindan and Tangmunarunkit [5] obtained a snapshot of the Internet topology by using the Mercator program. Mercator is designed to map the network from a single source without an initial database of target destination nodes for probing, but to the heuristically determined destination address space. Mercator also uses source routing to direct the hop-limited probes in directions other than radially from the sender. In this way Mercator discovers crosslinks that otherwise might not have been discovered. The authors have collected a large data set in 1999, resulting in a graph consisting of 228 263 nodes and 320 149 links.

As far as the node degree distribution is concerned, the results of Govindan and Tangmunarunkit [5] indicated that for node degree values below 30, the plot on a log–log

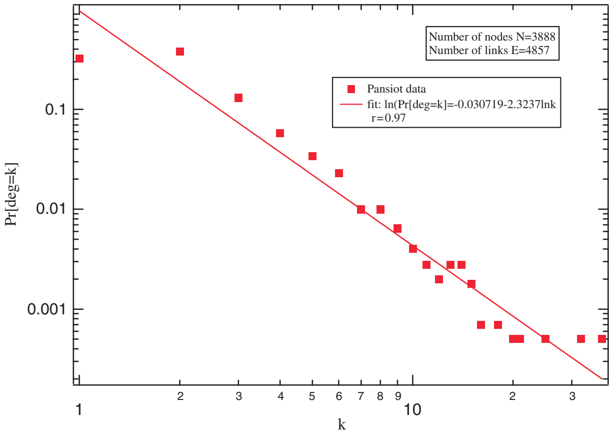

Figure 3. The node degree distribution based on data of Pansiot and Grad [10].

scale is linear, suggesting a power-law behaviour. However, the distribution becomes significantly more diffuse for node degrees larger than 30.

The Rocketfuel project [8] had the goal to map 10 diverse ISP networks. Publicly available *traceroute* servers have been used as sources, while the destinations have been chosen out of the BGP routing table, or out of RouteViews when BGP tables were not available. They further used direct probing techniques to obtain an as accurate as possible map, but with a limited number of measurements. Traceroutes that contribute most to the map are chosen, and the others are omitted, trading the accuracy for efficiency. After aliases have been resolved, the authors found that almost 70% of the routers had only one interface, 10% of the routers had two aliases, while even one router had 24 aliases.

In our analysis, we have used two sets of data from the Rocketfuel project. We have merged the tables for these 10 topologies together, and after eliminating duplicate nodes and links (links and nodes common to several ISPs), we created the aggregate topology with the size of 42 875 nodes and 103 681 links. The node degree distribution in the resulting graph is shown in Figure 4(a).

It is interesting to investigate how the alias resolving process affects the node degree distribution. We obtained the aggregate topology over all ISP's, counting 48 399 nodes and 88 437 links. The degree distribution is plotted in Figure 4(b). Figure 4 seems to indicate that the alias resolution process has no significant influence on the node degree distribution. Both distributions follow a power law, with a slightly different exponent.

4.2. *Node degree distributions based on CAIDA, RIPE NCC and PlanetLab data*

For our analysis, we have created maps of (a part of) the Internet using both CAIDA and RIPE measurement data. By merging all *traceroutes* as described in Section 2, a router-level map is obtained, from which the node degree distribution is computed and illustrated in Figure 5(a).

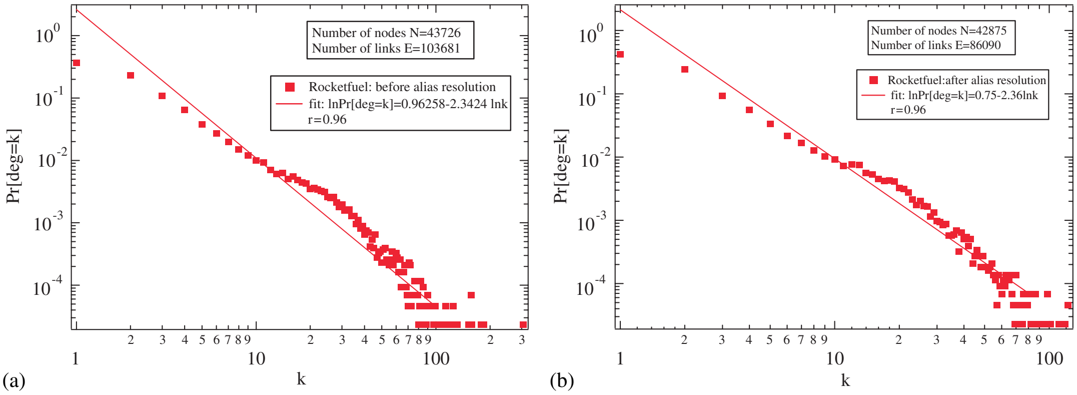

Figure 4. (a) The node degree distribution based on Rocketfuel data (January 2002) before alias resolution; and (b) the node degree distribution based on Rocketfuel data (January 2002) after alias resolution.

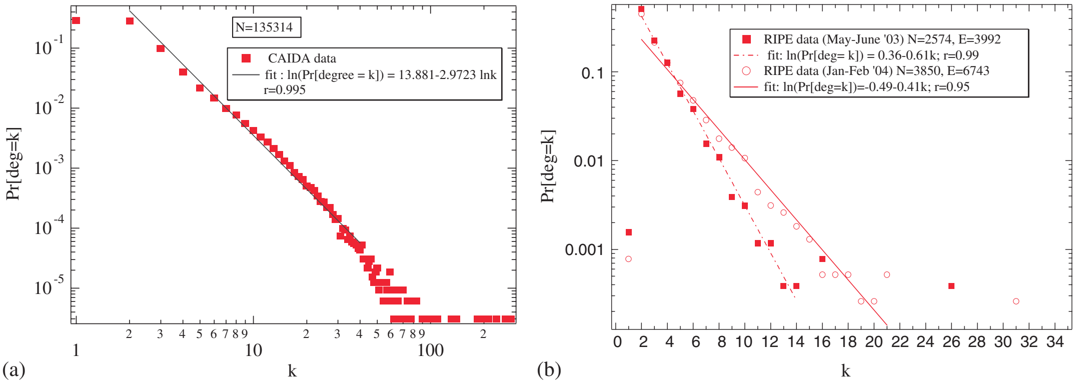

Figure 5. (a) The node degree distribution based on CAIDA data (1, 2, 3 April 2003); and (b) the node degree distribution based on RIPE NCC data.

Similarly, a map of the Internet has been constructed from RIPE measurement data by assembling the most dominant non-erroneous *traceroute* paths from each of x test boxes to all the other boxes in the period May–June 2003 (data set 1), and from each of y boxes to all the others in the period January–February 2004 (data set 2). In this way the graph named G_1 has been created, consisting of 2574 nodes and 3922 links, and 3850 nodes and 6743 links, in two different periods, respectively.** No effort has been made to determine the aliases, hence, the graph G_1 represents the approximation of the Internet interface map, not of the Internet router

**In our previous work [7] we have performed a similar study, based on data available in 2002.

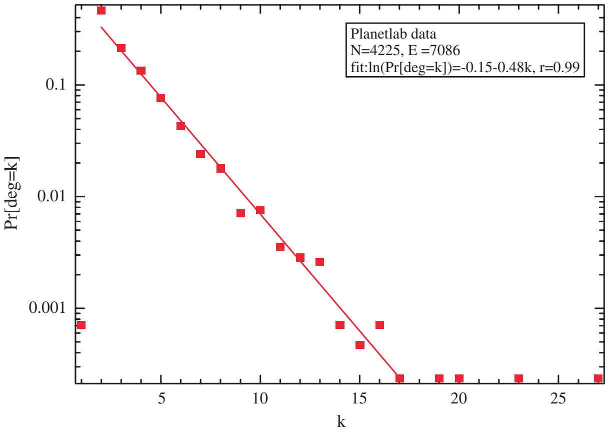

Figure 6. The node degree distribution derived from Planetlab measurements.

map. Figure 5(b) shows that the probability density function (pdf) of the node degree in both instances of the graph G_1 is exponentially decreasing over nearly the entire range. The exponential decay rate β decreased from 0.67 $\approx \ln 2$ in 2002 (see Reference [7]) to 0.61 in May–June 2003 and to 0.41 in January–February 2004, while gradually the number of nodes in G_1 increased over time.

This is a quite intriguing result since all published results based on many different measured data sets indicate that the degree distribution of the (sub)graph of the Internet should obey a power law. Our results suggests that when the observed subgraph of the Internet is small, no power law is observed. When gradually increasing the observed part, the decay rate β decreases and the exponential behaviour seems to turn into a power law. Another reason for the observed exponential node degree may lie in the fact that each RIPE measurement box acts as source as well as destination and that the RIPE measurement configuration mainly views the European part of the Internet.

These observation and conclusion seem to be confirmed by the results of measurements on the PlanetLab network as well. By merging the *traceroutes* from each of 79 nodes to all the other, a topology consisting of 4226 nodes and 7171 links was produced. No alias resolution technique has been implemented. The node degree distribution in this map has been computed, and is illustrated in Figure 6.

Again, a higher quality fit has been achieved on a log–lin than on a log–log scale, even though the slope coefficient takes the value 0.48.

4.3. The node degree distribution in multicast

To the best of our knowledge, only few results have been published on the characteristics of multicast routing trees. The first one has been provided by Chalmers and Almeroth [13], who have looked into the properties of the Internet multicast trees on Mbone. They have gathered multicast tree data for four live multicast sessions: the 43rd IETF meeting in December 1998 and the NASA shuttle launch in February 1999, each of them consisting of a separate audio and video channel. The path from each receiver to the source has been traced via *mtrace* (multicast *traceroute*) [14]. However, since receivers are traced one after another, the receivers participating for a short time may have been missed. Indeed, only 43% of receivers for IETF and 29% for NASA have been successfully traced. The *mtrace* data has been used for each *data set* to reconstruct a multicast tree. Chalmers and Almeroth have developed the tool *mwalk*, that builds an activity graph of all possible trees over time: the whole session has been divided in 10 000 intervals, and to each receiver an activity table is assigned, with the time intervals in which that particular receiver was a member of the group. In Figure 7 we have plotted the node degree distribution for one realization of the tree, in 43rd IETF video *data set*, when 129 receivers have been traced to belong to the group.

After fitting their data on different scales, we noticed that the best fit of all (with the correlation coefficient of 0.91) has been obtained for the linear fit on the log–lin scale, suggesting rather exponentially than polynomially distributed node degrees. Interestingly, the slope of the curve in Figure 7 is approximately the same as that observed in Figure 5(b) of RIPE.

The only other result on the multicast tree degree distribution so far has been provided by Dolev *et al.* [15]. They have investigated properties of multicast trees obtained from Internet measurement data. The data of their multicast analysis has been obtained via unicast *traceroute* measurement. They have used two *data sets*: first, on the underlying topology provided from a mapping project presented in Reference [9] (using *traceroute* measurements), they generated SPTs using the Dijkstra algorithm. The second *data set* has been created based on *traceroute* measurements of the paths between the root and the clients in the client population of www.bell-labs.com. Dolev *et al.* [15] do not state in their paper whether loops occurred in their data, nor how they approached and treated that phenomenon. In Reference [15, Figures 6 and 7] they have plotted the node degree distributions in both *data sets* on a log–log scale, and fitted with the linear function decaying with the rate -3.40 and -3.18 for the first and the second *data set*, with the correlation

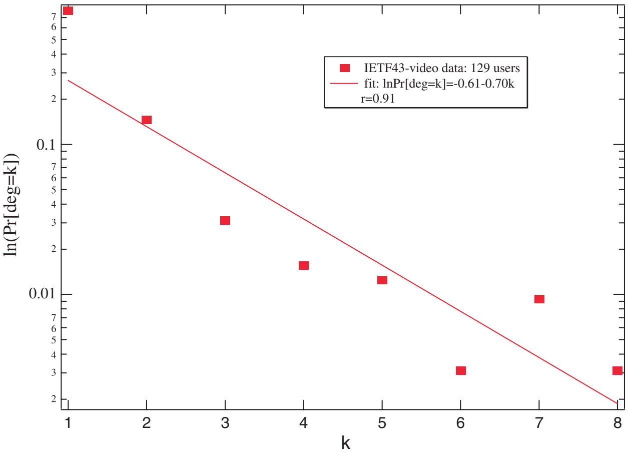

Figure 7. The node degree distribution in IETF 43rd meeting-video data set (7–11 December 1998) from Reference [13].

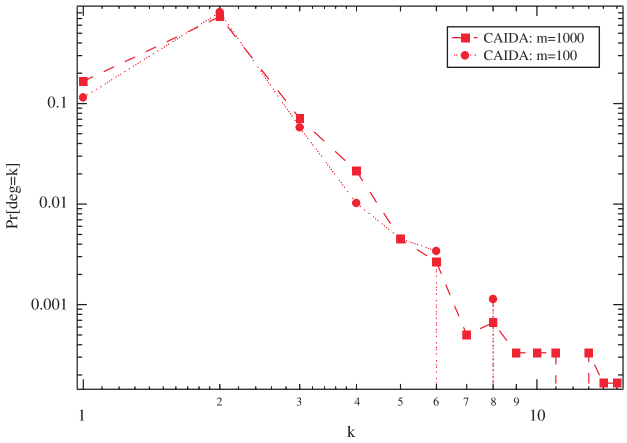

Figure 8. The degree distribution in multicast tree based on CAIDA traceroute data ($m = 1000$ and 100).

coefficients[††] of 0.9897 and 0.9829, respectively. These findings seem to suggest a power-law structure of the node degree distribution in multicast routing trees, which contradicts that of Figure 7.

4.4. Comparing the node degree distribution of trees in RIPE and CAIDA

In order to understand the discrepancy in Section 4.3, we have further investigated the node degree distribution obtained from the Skitter project. We present the resulting distributions only for trees rooted at the source in the United States, since the distributions for trees rooted at two other sources are almost identical. Our results indicate that based on CAIDA *traceroute* data, for $m \geqslant 100$, as given in Figure 8 ($m = 1000$ and 100), the node degrees seem to be polynomially distributed.

When we plotted the node degree distribution for $m = 50$ on two different scales, log–lin in Figure 9(a), and log–log scale in the inset in Figure 9(a), we noticed one remarkable property: when fitting the data with a linear function in both scales, the quality of the fit seems to be comparable! This can be seen in the value of the linear correlation coefficients r_α and r_β, that represent the measure of the quality of fit. The resulting slope coefficient α and the correlation coefficient r_α for linear fits on log–log scale, and the slope coefficient β and the correlation coefficient r_β for linear fits on log–lin scale, for $m = 50$ destinations derived from CAIDA *traceroute* data, are summarized in Table II(a). We notice that the quality of the fit on a log–log scale for $m = 50$ is only slightly higher than the quality of the fit on the log–lin scale. Based on the given data, it is questionable whether it is reasonable to claim that the degree distribution of the union of *traceroute*s representing a multicast routing tree follows a power-law distribution for small m.

Even more doubt is raised after plotting the data obtained from RIPE in two different scales, for $m = 20$ and 50. Although fitting the data on a log–log scale is somewhat better,

[††] The linear correlation coefficients measure the extent of linear relationship of two variables, and are given by $r = \frac{\mathrm{cov}(y,x)}{\sqrt{\mathrm{var}(y)\mathrm{var}(x)}}$.

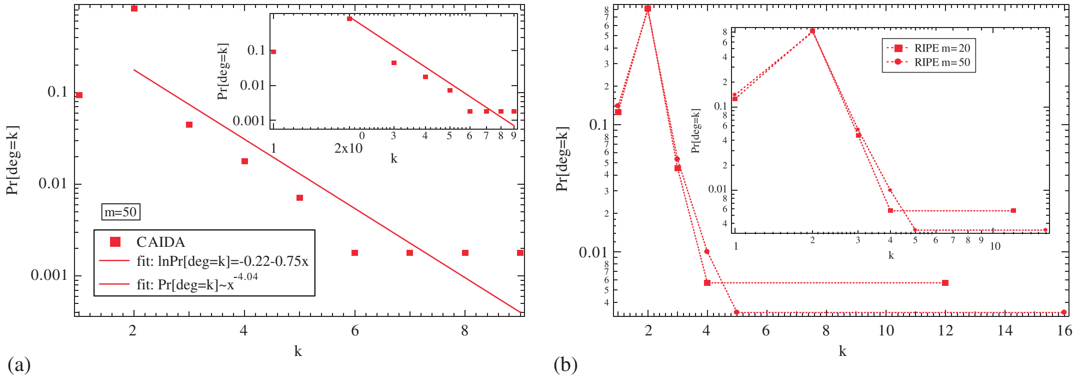

Figure 9. (a) The degree distribution in multicast tree based on CAIDA traceroute data ($m = 50$) on log–lin scale and log–log in the inset; and (b) the degree distribution in multicast tree based on RIPE traceroute data ($m = 50$ and 20) on log–lin scale and log–log in the inset.

Table II. (a) Slope and correlation coefficients for linear fits on log–log and log–lin scale in Caida data ($m = 50$). (b) Correlation coefficients for linear fits on log–log and log–lin scale in RIPE data ($m = 50$ and 20).

(a)		(b)			
$m = 50$		$m = 50$		$m = 20$	
α	4.04	r_α	0.78	r_α	0.74
r_α	0.95	r_β	0.61	r_β	0.64
β	0.75				
r_β	0.93				

the quality of fitting with linear functions in both scales is considerably deteriorated, as can be seen from Table II(b) and Figure 9(b). Therefore, we conclude that the degree distribution in the multicast tree for small m does not convincingly follow a power law. For larger values of the number of destinations m the degree distribution seems to follow a power law, when multicast trees are created as union of *traceroutes*. The result of Chalmers and Almeroth implies that if another method for constructing trees is used other then the union of *traceroute* paths, power laws might not be observed for even larger values of m.

5. THEORY, MEASUREMENT AND SIMULATIONS OF MULTICAST TREES

5.1. The number of links in the SPT

The average number of links used in multicast has been regarded as a measure for the gain of multicast over unicast. Chuang and Sirbu [16], Phillips *et al.* [17], Chalmers and Almeroth [13] and Van Mieghem *et al.* [18] have all assumed multicast delivery along SPTs rooted at source to

m destinations uniformly chosen out of N nodes. This assumption has been confirmed by Internet measurements [13, 17]. Let us denote by $H_N(m)$ the number of hops or links in the SPT rooted at a particular source to m randomly chosen destinations, then the gain of multicast is expressed as $g_N(m) = E[H_N(m)]$.

Van Mieghem *et al.* [19] have shown that the SPT in the complete graph is exactly, and in the class $G_p(N)$ asymptotically, a uniform recursive tree (URT). A URT of size N is a random tree rooted at some source node and where at each stage a new node is attached uniformly to one of the existing nodes until the total number of nodes is equal to N. Although the random graphs $G_p(N)$ are not a good model for the Internet topology, the URT seems a reasonable accurate model for trees in the Internet [18]. For the URT, the average number of hops to m randomly chosen destinations, for every N and m is given by

$$g_N(m) = E[H_N(m)] = \frac{mN}{N-m} \sum_{k=m+1}^{N} \frac{1}{k} \tag{1}$$

In addition to the average (1), the exact probability generating function and probability distribution $\Pr[H_N(m) = k]$ is derived in Reference [4], from which the variance follows as

$$\mathrm{var}[H_N(m)] = \frac{N-1+m}{N+1-m} E[H_N(m)] - \frac{(E[H_N(m)])^2}{(N+1-m)} - \frac{m^2N^2}{(N-m)(N+1-m)} \sum_{k=m+1}^{N} \frac{1}{k^2} \tag{2}$$

The interest of this result as shown in Reference [4] is that for all $m = o(\sqrt{N})$, the normalized random variable

$$H_N^*(m) = \frac{H_N(m) - E[H_N(m)]}{\sqrt{\mathrm{var}[H_N(m)]}}$$

converges to a standard normal (Gaussian) random variable when $N \to \infty$.

In addition, we have verified our analytically derived results with the measurements on Internet. In Figure 10 the number of links has been plotted, for each source, as a function of m. In addition, for $N = 135\,314$ (the number of nodes in the Internet map derived from *traceroute* measurements) and for various values of m (in the range [50, 20 000], the values of the functions $E[H_N(m)]$ and $E[H_N(m)] \pm 6\sigma_N(m)$ where the standard deviation $\sigma_N(m) = \sqrt{\mathrm{var}(H_N(m))}$ are also plotted in the same Figure 10. Since $\sigma_N(m)$ is much smaller than $g_N(m)$, the number of links in the URT is well approximated by the mean, $H_N(m) \approx E[H_N(m)]$, for large values of N. The measured number of links falls in the range $E[H_N(m)] \pm 6\sigma_N(m)$, indicating again that the URT models the multicast trees reasonably well.

5.2. Simulations of the union of SPTs in the complete graph

If we assume that *traceroutes* represent shortest paths, than the subgraph of Internet graph G_1 in Section 4.2 can be modelled as a union of shortest paths. We now compare the node degree distribution in the union of SPTs in a complete graph K_N with uniformly distributed link weights with the RIPE data, for small values of m compared to N. We have performed the following set of simulations: in a complete graph K_N with uniformly distributed link weights consisting of $N = 1000$ nodes, m nodes have been chosen randomly, where $m = 3, 5, 10, 20, 50, 100, 200, 500, 700, 1000$. For a particular m, the SPT rooted in each of m nodes to the other $m-1$ destinations has been computed as well as their union, from which the node degree distribution has been calculated. For each m, 10 000 iterations have been performed. The

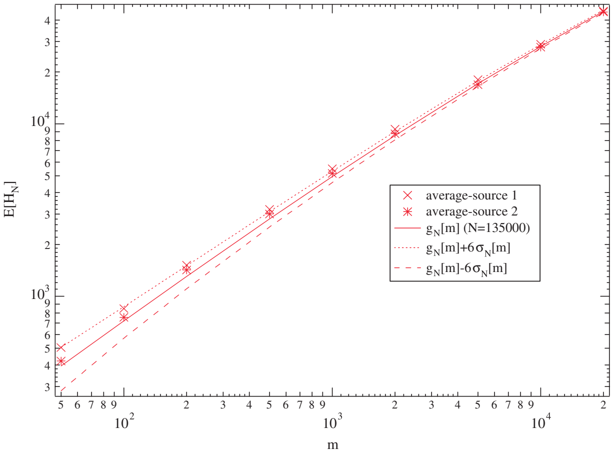

Figure 10. The average number of links (Caida measurements and theoretical value).

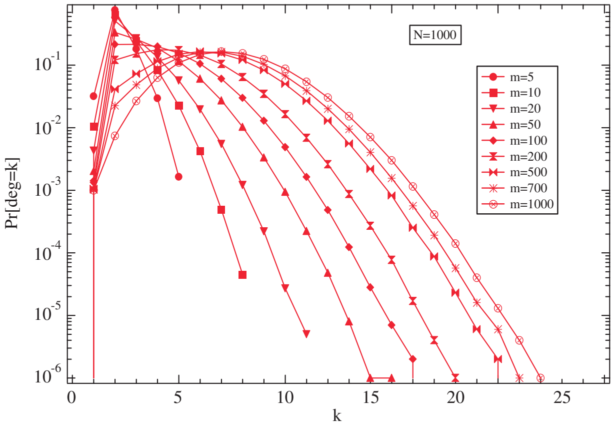

Figure 11. The node degree distribution in the union of shortest path trees in K_N $(N = 1000)$.

resulting distributions are presented in Figure 11. The correspondence with the RIPE measurement configuration is that m reflects the number of test boxes (which is growing over time) while N represents the total number of observed routers in the union of traceroutes. Figure 11 reveals that for small m compared to N the simulated pdf of node degrees resembles the shape observed from RIPE data, except for several outliers in Figure 5(b).

Finally, the ratio of the average of the number of nodes with degree k, denoted by D_k^N, over the total number of nodes in the URT obeys for large N, is computed in Reference [20, Chapter 16] as

$$\frac{E[D_k^N]}{N} = \frac{1}{2^k} + O\left(\frac{\log^{k-1} N}{N^2}\right) \tag{3}$$

which is, for large N, close to $\Pr[\deg = k]$, the probability that an arbitrary node has a degree k. Hence, the exponential decay rate β of the pdf of the node degree in the URT equals $\beta_{\text{URT}} = \ln 2 = 0.693$. Figure 5(b) shows that the measured β approaches β_{URT} if m decreases. In that case, the union of all shortest paths seems close to a URT explaining the observed exponential degree distribution.

5.3. *The weight of the SPT*

van der Hofstad *et al.* [4] have derived the average $E[W_N(m)]$ of the sum of the weights $W_N(m)$ in the SPT to m uniform multicast users in the random graph $G_p(N)$ with exponentially distributed link weights,

$$E[W_N(m)] = \sum_{j=1}^{m} \frac{1}{N-j} \sum_{k=j}^{N-1} \frac{1}{k} \leqslant \frac{\pi^2}{6} \tag{4}$$

but no analytic expressions are known for the distribution $\Pr[W_N(m) \leqslant x]$ nor for the corresponding probability generating function $\varphi_{W_N(m)}(z) = E[\exp(-zW_N(m)]$. Here, we complement the analytical results derived in Reference [4]. The significance of $W_N(m)$ for multicast is that $W_N(m)$ can represent the cost of used resources of the multicast tree, defined as the sum over all links in the multicast tree of the (monetary) costs of the resources used per link.

We confined ourselves to complete graphs with exponentially distributed link weights with mean 1. For each number of nodes N, 10^5 topologies were generated randomly. For each of these topologies, $m \in \{1, N-1\}$ nodes were uniformly chosen. The SPT from an arbitrary node to m uniform other nodes and the SMT connecting m nodes have been computed. The SPT is computed by using the Dijkstra algorithm, with $N \leqslant 100$. Depending on m, the SMT [2] is generated using different algorithms. For $m = 2$, the SMT problem reduces to the computation of the shortest path between those two users. If $m = N$, the SMT is actually the (complete) minimum spanning tree, and is computed with the Prim algorithm. For $2 < m < N$, the SMT problem belongs to the class of *NP-complete* problems. Certain reductions [2] in the topology decrease the number of nodes and links and increase the speed of simulations in that reduced graph. In spite of the implemented reductions, the simulation process is nevertheless extremely time consuming for large N. Therefore, we restrict the simulations of SMT to graphs with $N \leqslant 20$. In each graph and for each m, the sum of the weights as well as the number of links in both the SPT and the SMT have been stored in 4 histograms. From these histograms, the pdf of the sum of the link weights $f_{W_N(m)}(x) = (\mathrm{d}/\mathrm{d}x)\Pr[W_N(m) \leqslant x]$ in the SPT and the SMT have been deduced.

Figure 12 gives the pdf of the sum of the link weights $f_{W_N(m)}(x)$ in the SPT.

The average value $E[W_N(m)]$ of the sum of the link weights in the SPT and the SMT is plotted as a function of the multicast group size m, for the number of nodes $N = 20$ in Figure 13(a). Apart from the match between simulations and theory for the SPT, this figure reveals that $E[W_N(m)]$ for the SMT seems similar (apart from some scaling factor) to that of the corresponding SPT. In Figure 13(b), simulation results of the variance of $W_N(m)$ in the SPT and the SMT are shown. So far, $\text{var}[W_N(m)]$ has not been derived analytically (except for $m = N-1$ in Reference [20, Chapter 17]).

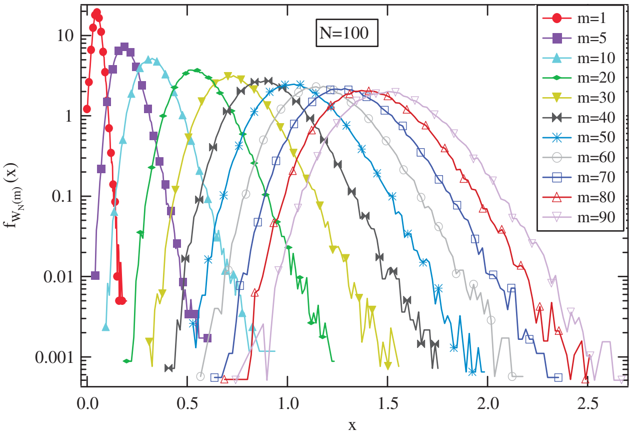

Figure 12. The pdf of sum of the weights in the SPT for $N = 100$.

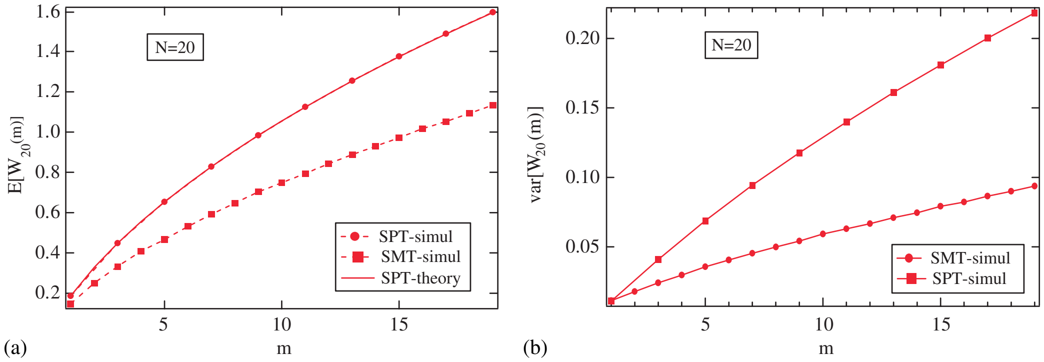

Figure 13. (a) The average value of the sum of the link weights for SPT and SMT ($N = 20$); and (b) the variance of the sum of the link weights for SPT and SMT ($N = 20$).

Although N is small (which allows us to show the entire m-range), Figure 14 indicates that the scaled random variable

$$X_N(m) = \frac{W_N(m) - E[W_N(m)]}{\sqrt{\mathrm{var}[W_N(m)]}}$$

is close to a Gumbel type $\mathrm{e}^{-\mathrm{e}^{-x}}$, which may suggest, for all m, that

$$\lim_{N\to\infty} \Pr[X_N(m) \leqslant x] = \mathrm{e}^{-\mathrm{e}^{-(\pi\sqrt{6})x-\gamma}} \tag{5}$$

where $\gamma = 0.5772...$ is Euler's constant. For the particular case of $m = 1$, we are able to prove this result [21]. However, simulations for larger $N > 1000$ seem to indicate that $X_N(m)$ tends to a

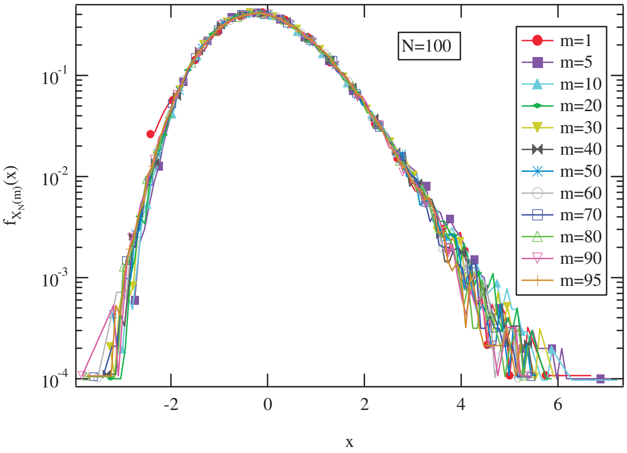

Figure 14. The pdf of the scaled random variable $X_N(m) = \frac{W_N(m) - E[W_N(m)]}{\sqrt{\text{var}[W_N(m)]}}$.

normalized Gaussian for $m > 1$. As a matter of facts, for $m = N - 1$, convergence of $X_N(N-1)$ to a normalized Gaussian can be proved.[‡‡] Hence, $X_N(m)$ converges only slowly towards its asymptotic limit implying that simulations are not the best device to obtain the asymptotic distribution.

6. CONCLUSIONS

Several ambiguities related to *traceroute* data have been discussed, such as inaccuracies, alias resolving and bias sampling, that make the reliability of derived conclusions on the topological characteristics of Internet questionable.

The detailed review of Internet map node degree distribution has been provided. The discrepancy in the node degree distributions based on RIPE *traceroute* data with results obtained by others, seems to confirm the above-stated observation on reliability of *traceroutes*.

We have further analysed the node degree distributions in multicast trees, created as the union of *traceroutes*. The scarce results on node degree distributions in multicast trees have been controversial as well. Our results on the node degree distribution in multicast trees seem to suggest that based on *traceroute* data, universal power-law behaviour cannot be claimed. While for larger values of the number of destinations m the degree distribution seems to follow power law, for $m \leqslant 50$ based on both RIPE and CAIDA *traceroute* data this does not seem to be the case. Internet measurements seem to suggest that for the small number of users m , the URT represents a reasonable model for multicast trees in Internet. In this paper

[‡‡] R. van der Hofstad, G. Hooghiemstra and P. Van Mieghem, The weight of the shortest path tree, unpublished.

we proposed analytical expressions (1) and (4) for assessing the gain/cost of the SPT. The cost of the SPT has been compared to the cost of SMT. The measurement results indicated that the average number of links in multicast trees lies in the range $g_N(m) \pm 6\sigma_N(m)$ where $\sigma_N(m) = \sqrt{\mathrm{var}(H_N(m))}$ for all values of m. Since $\sigma_N(m)$ is much smaller than $g_N(m)$ for large N, the number of links in the URT is well approximated by the mean, $H_N(m) \approx E[H_N(m)]$.

ACKNOWLEDGEMENTS

We would like to thank Robert Chalmers (University of Santa Barbara), Neil Spring (University of Washington), Andre Broido and KC Claffy (CAIDA) and Henk Uijterwaal (RIPE NCC) for providing us with their measurement data.

REFERENCES

1. Stevens WR. *TCP/IP Illustrated*, *volume 1*, *The Protocols*. Addison-Wesley: Reading, MA, 1994.
2. Hwang F, Richards D, Winter P. *The Steiner Tree Problem* (*Annals of Discrete Mathematics, vol. 53*). North-Holland: Amsterdam, 1992; 103.
3. Van Mieghem P, Janic M. Stability of a multicast tree. *Proceedings of IEEE INFOCOM*, New York, NY, U.S.A., July 2002.
4. van der Hofstad R, Hooghiemstra G, Van Mieghem P. Size and weight of shortest path trees with exponential linkweights. *Combinatorica*, *Probability and Computing*, to appear.
5. Govindan R, Tangmunarunkit H. Heuristics for Internet map discovery. *Proceedings of IEEE INFOCOM*, Tel Aviv, Israel, March 2000.
6. Yao B, Viswanathan R, Chang F, Waddington D. Topology inference in the presence of anonymous routers. *Proceedings of IEEE INFOCOM*, San Francisco, CA, U.S.A., 2003.
7. Janic M, Kuipers F, Zhou X, Van Mieghem P. Implications for QoS provisioning based on *traceroute* measurements. *Proceedings of 3rd International Workshop on Quality of Future Internet Services*, *QofIS2002*, Zurich, 2002.
8. Spring N, Mahajan R, Wetherall D. Measuring ISP topologies with Rocketfuel. *Proceedings of ACM SIGCOMM*, Pittsburgh, PA, U.S.A., August 2002.
9. Burch H, Cheswick B. Mapping the Internet. *IEEE Computer* 1999; **32**:97–98, 102.
10. Pansiot J, Grad D. On routes and multicast trees in the Internet. *ACM Computer Communication Review* 1998; **28**(1):41–50.
11. Lakhina A, Byers J, Crovella M, Xie P. Sampling biases in IP topology measurements. *Proceedings of IEEE INFOCOM*, San Francisco, CA, U.S.A., April 2003.
12. Bollobas B. *Random Graphs*. Academic Press: New York, 1985.
13. Chalmers R, Almeroth K. On the topology of multicast trees. *IEEE Transactions on Networking* 2003; **11**(1): 153–165.
14. Fenner W, Casner S. A 'traceroute' facility for IP multicast. *Technical Report*, draft-ietf-idmr-traceroute-ipm-*txt, Internet Engineering Task Force (IETF), August 1998.
15. Dolev D, Mokryn O, Shavitt Y. On multicast trees: structure and size estimation. *Proceedings of IEEE INFOCOM*, San Franscisco, CA, U.S.A., April 2003.
16. Chuang JC-I, Sirbu MA. Pricing multicast communications: a cost based approach. *Proceedings of INET*, Geneva, Switzerland, 1998.
17. Phillips G, Shenker S, Tangmunarunkit H. Scaling of multicast trees: comments on the Chuang–Sirbu scaling law. *Proceedings of SIGCOMM'99*, Cambridge, MA, U.S.A., August 1999.
18. Van Mieghem P, Hooghiemstra G, van der Hofstad R. On the efficiency of multicast. *IEEE/ACM Transactions on Networking* 2001; **9**(6):719–732.
19. Van Mieghem P, Hooghiemstra G, van der Hofstad R. A scaling law for the hopcount in Internet. *Report 2000125* (http://www.nas.ewi.tudelft.nl/people/ Piet/teleconference.html)
20. Van Mieghem P. *Performance Analysis of Communications Systems and Networks*. Cambridge University Press: Cambridge, 2005.
21. Janic M, Van Mieghem P. The gain and cost of multicast routing trees. *International Conference on Systems*, *Man and Cybernetics* (*IEEE SMC 2004*), The Hague, The Netherlands, 10–13 October, 2004.

AUTHORS' BIOGRAPHIES

Milena Janic graduated from the University of Belgrade, Belgrade, Serbia and Montenegro, in 1999. She is currently pursuing the PhD degree in the Network Architectures and Services Group, at the Delft University of Technology (TUDelft), Delft, The Netherlands. Her research interest include modelling and optimization of the architecture and performance of IP multicast and peer-to-peer networks.

Piet F. A. Van Mieghem is professor at the Delft University of Technology with a chair in telecommunication networks and chairman of the basic unit Network Architectures and Services (NAS). His main research interests lie in new Internet-like architectures for future, broadband and QoS-aware networks and in the modelling and performance analysis of network behaviour and complex infrastructures. Professor Van Mieghem received a Master and PhD in Electrical Engineering from the K. U. Leuven (Belgium) in 1987 and 1991, respectively. Before joining TUDelft, he worked at the Interuniversity Micro Electronic Center (IMEC) from 1987 to 1991. During 1993 to 1998, he was a member of the Alcatel Corporate Research Center in Antwerp where he was engaged in performance analysis of ATM systems and in network architectural concepts of both ATM networks (PNNI) and the Internet. He was a visiting scientist at MIT (1992–1993) and, in 2005, he was visiting professor at UCLA. Currently, he is member of the editorial board of the journal Computer Networks.