# NGCE - Network Graphs for Computer Epidemiologists

Vasileios Vlachos[1], Vassiliki Vouzi[2],
Damianos Chatziantoniou[1], and Diomidis Spinellis[1]

Department of Management Science and Technology,
Athens University of Economics and Business (AUEB),
Patission 76, GR-104 34, Athens, Greece
[1] {vbill, damianos, dds}@aueb.gr
[2] vavouzi@dmst.aueb.gr

**Abstract.** Graphs are useful data structures capable of efficiently representing a variety of technological and social networks. They are therefore utilized in simulation-based studies of new algorithms and protocols. Inspired by the popular TGFF (Task Graphs For Free) toolkit, which creates task graphs for embedded systems, we present the NGCE, an easy to use graph generator that produces structures for the study of the propagation of viral agents in complex computer networks.

**Keywords:** Network graphs, Computer epidemiology, Malicious software

**Designated track:** Computer Security

## 1 Introduction

The design and evaluation of robust network protocols and algorithms is a difficult and tedious problem, mainly due to the effort and the economic cost required to deploy a prototype system in a large scale network in order to test the proposed designs. Moreover, some cases cannot conceivably be studied *in vivo*. For example, when studying malicious software or biological infectious agents, it is not possible to examine the spread of worms or viruses simply by releasing live viral code to the Internet or virulent strains to the society. The scientific community addresses these problems primarily by making extensive use of simulations.

The most accurate results come from simulations that utilize real-data topologies, but since most of the time these data are not available [1], synthetic graphs are used instead. Unfortunately the creation of artificial graphs imposes an additional effort, because they have to be built from scratch. Furthermore, as a result of the burden of constructing graph structures, researchers are forced to build only a small number of them in order to study their ideas. We believe that the existence of an easy to use graph generator will help the scientific community to evaluate the proposed algorithms under a broader scope using multiple topologies. This approach would offer the opportunity to gain insights on the weighted algorithms and to uncover their strengths and weaknesses

P. Bozanis and E.N. Houstis (Eds.): PCI 2005, LNCS 3746, pp. 672–683, 2005.

in a much finer granularity than by using a single example or a limited number of graphs. The paper [2] showed the impact of graph generators in simulation outcomes, and [3] pointed out the correlation between the physical infrastructure and the problem under investigation during the design of graph models. Most of the recent research related to the generation of network graphs has focused on routing, resource preservation, and performance problems. On the contrary, to the best of our knowledge, little attention has been paid to the development of graph generation tools specially crafted for the study of the spread of malicious applications.

Another important issue concerns the difficulties that different research groups face in comparing their results because of the absence of a standard set of graphs or an application capable of constructing reproducible graphs. Given that it is quite difficult to generate a single set of graphs large enough to cover all needs, our goal was to introduce a tool that could regenerate graphs with predefined and repeatable properties. We focus on malcode propagation dynamics, because we believe that it represents a situation where alternative study methodologies, besides the use of simulations, are limited and not acceptable. Moreover, some studies take place during the outbreak of an epidemic, leaving insufficient time for the preparation of various graph topologies. We believe that this kind of support tool will be useful to speed up the research on the spread of viral code, especially during crisis situations. The embedded systems design community benefited a lot from the existence of tools such as the TGFF [4], which are able to generate specially crafted reproducible task-graphs for embedded systems. We hope that similar gains will also arise for the worm containment community.

Our decision was to incorporate within this application the most common topologies that have been and are still widely used to simulate the virulent activity of malware. We presume however, that this tool may also be useful for other studies related to social and technological networks as it is quite easy to add new topologies by developing appropriate plug-ins.

The rest of this paper is organized as follows. In section 2 we survey the most representative network topologies and we present their applications in a number of different circumstances. In section 3 we summarize the available graph generators. Section 4 describes the design of our tool and discusses various aspects of its implementation, while section 5 analyzes the graphs produced by our tool and comment on their characteristics. In section 6 we present possible use-case scenarios for the NGCE tool. Finally section 7 concludes this paper.

## 2 Network Graph Topologies

Although graphs have been widely used thoroughly to study a variety of interesting theoretical problems, only a small percentage of them are appropriate for network related problems and have been utilized in this scope. The most suitable topologies for this type of research are homogenous graphs, random graphs, random graphs with fixed connectivity, scale-free graphs and some variations of them that we will discuss below.

### 2.1 Homogeneous Graphs

Homogeneous or fully connected graphs were for many years the epidemiologists' preferred choice for describing the spread of infectious diseases. This topology has been recently adapted to model the growth of computer viruses and worms [5,6,7,8,9,10,11,12,13,14,15]. NGCE supports homogeneous graphs because they offer significant advantages. First, analytical mathematical models can be easily applied to them; second, they provide a good abstraction of very large networks when the majority of the susceptible hosts are accessible from an infectious agent and third, performing simulations using a homogeneous graph and comparing their outcomes with the mathematical analytical results is an excellent way to ensure that implementation details did not corrupt the simulation model. Homogeneous graphs can be built as follows.

1. Start with $N$ vertices and no edges.
2. Connect each vertex with all the other vertices.

A homogeneous graph with $N$ nodes will always end up with $(N^2 - N)/2$ edges.

### 2.2 Random Graphs

Until recently, it was believed that random graphs provide a good approximation to very large networks such us the Internet. Barabási et al [16] proved however, that the connectivity of the Internet, along with that of many other technical and social networks, obeys a power law distribution forming scale-free graphs. Prior to these findings, a large number of simulations that investigated the spread of malicious code, had been performed on random graphs [6,7,8,17,15]. Due to this fact, and, more importantly, in order to allow the comparison between older and current studies, we decided to include them in our tool. Since numerous algorithms have been proposed to construct random graphs, we selected to implement the most widely adopted one, in particular the Erdős-Rény algorithm [18] or the Gilbert algorithm according to others [19]. The algorithm works as follows.

1. Start with a graph with $N$ nodes and no edges.
2. Connect each pair of two nodes with probability $P_{er}$.

This results in an Erdős-Rény graph. The shape of the connectivity distribution however is highly dependent on the value of the $P_{er}$ probability.

### 2.3 Scale-Free Graphs

Scale-free structures exist in a stunning range of heterogeneous systems ranging from biological and social to purely technological [20] networks such as the World Wide Web (WWW) [21,22,23], the physical connectivity of the Internet [24,25] or the network of people connected by e-mail [26]. Studies have explored the spread of malicious code in scale-free computer networks with interesting but

conflicting results [27,15,28] while [29] investigated the resilience of the Internet to random breakdowns. We assume that, in light of these recent evidences, simulation research will increasingly be based on scale-free topologies and thus we support them in the NGCE tool. Barabási and Albert demonstrated that the creation of scale-free networks should include two definitive characteristics: *Incremental growth* and *preferential connectivity*. *Incremental growth* is the process of adding new nodes to an existing graph. *Preferential connectivity* or preferential attachment describes the tendency of a newly added node to be connected with highly connected nodes. The elimination of either of these properties lead to graphs with temporal scale-free characteristics. Under certain circumstances it is possible to construct scale-free graphs without enforcing growth and preferential connectivity simultaneously. We felt that the inclusion of a flexible model that allows the experimentation with either of these modes or a combination of them would leverage the development of a larger variety of scale-free graphs.

**Preferential connectivity.** In order to construct a BA (Barabási and Albert) graph working with a fixed number of nodes without the incremental growth property, but with the preferential connectivity, we have to use a limited number of edges [30]. The algorithm consists of the following phases.

1. If the number of edges is less than the number of nodes, select randomly a vertex and connect it with probability
$$P(k_i) = \frac{k_i}{\sum_j k_j} \tag{1}$$
to the vertex $i$.
2. Repeat step 1.

If the number of edges is approximately equal to the number of nodes the constructed graph exhibits scale-free characteristics.

**Incremental Growth.** One, but not sufficient ingredient, of the scale-free networks is the incremental growth property, but without the preferential connectivity option; as new nodes are added to the graph the scale-free nature of the network diminishes. We thought that it would be best, if we enabled the construction of graphs based solely on the incremental growth for further study and experimentation. The algorithm has the following structure.

1. Create a pool $K$ and add to it $m_o$ initial nodes.
2. Create a pool $L$ of all the other nodes.
3. Remove randomly a node $i$ from the pool $L$ and connect it to a randomly chosen node from the pool $K$.
4. Add node $i$ to the pool $K$.
5. While the pool $L$ is not empty, repeat step 3.

It is also possible to connect a node with $m > 1$ other nodes from pool $K$ as long as the size of the pool $K$ is larger than m.

**Complete Model.** By combining both of the incremental growth and preferential connectivity properties our system will approximate the BA model as closely as possible. The algorithm is constituted by two major parts. For the first $m_o$ nodes:

1. Create a pool $K$ and add to it $m_o$ initial nodes.
2. Create a pool $L$ of all the other nodes.
3. Remove randomly a node l from the pool $L$ and connect it to a randomly chosen node from the pool $K$.
4. Add node l to the pool $K$.

And for the rest of the nodes, we follow closely the BA's algorithm.

1. Remove randomly a node $i$ from the pool $L$.
2. Select randomly a vertex from the pool $K$ and connect it with probability

$$P(k_i) = \frac{k_i}{\sum_j k_j} \tag{2}$$

   to the vertex $i$.
3. Add $i$ to the pool $K$.

The first part of the algorithm is arbitrary since Barabási and Albert, did not describe how to connect the $m_o + 1$ node.

To verify that the produced graphs obey the power law distribution necessary to characterize a graph as scale-free, we developed scripts to parse the constructed graph and extract the connectivity probabilities in a *gnuplot* format file.

### 2.4 Random Graphs with Fixed Connectivity

Random graphs with fixed connectivity constitute a nontrivial network topology that is often encountered in grid systems. Considering the significant importance of such systems and the fact that older studies have been based on them [6,8], we also included these structures in our NGCE tool. We implemented a custom algorithm for the creation of random graphs with fixed connectivity. In our approach, the user chooses the number of nodes and the number of edges each node should contain. Then the algorithm works as follows.

1. Create an unconnected graph with $N$ nodes.
2. Add all nodes to a pool $L$.
3. Pick randomly a vertex $v$ and a vertex $u$ from the pool $L$, if the pool $L$ has at least two elements, else exit.
4. Connect vertex $v$ with vertex $u$ and decrease their available connectivity by one.
5. If vertex's $v$ or vertex's $u$ available connectivity is 0, remove it from the pool $L$.
6. Repeat step 3.

### 2.5 Custom Graphs

Sometimes it is necessary to measure the effects of various algorithms in non-standard graphs. We therefore added an option to our system that gives the ability to an experienced user to create non-typical graphs with custom properties based either on the random graph algorithm or on the scale-free algorithm.

## 3 Network Graph Generators

The recent evidence from Barabási and Albert regarding the scale-free nature of the World Wide Web and the Internet [24] raised a number of issues concerning the structure of other technological networks. Several researchers proposed algorithms that provide improved models for the representation of these networks or uncover insufficiencies of the current models, such as [31,32] for the Internet or [22] for the World Wide Web. Despite these interesting advances, little effort has been put into developing tools to automate the creation of graphs in an efficient and reproducible way.

The Georgia Tech Internetwork Topology Models (GT-ITM)[33] is a widely used tool to generate random and hierarchical graphs that model the topological structure of networks. GT-ITM requires some knowledge of graph theory and is operable only via the command line. Furthermore, the types of the constructed graphs are more suitable for studying network-related problems (routing, resource reservation), rather than the spread of malicious software. Many of these concerns have been addressed by [34], which added visualization capabilities and optimized the graph creation algorithms. The GT-ITM work however is still focused on routing policies.

Inet[35] is an effective tool that aims to construct Internet topologies. While it produces sufficiently good results that approximate closely the actual Internet structure, it works only in the Autonomous Systems (AS) level leaving out finer-grain topologies, as the IP connectivity. Furthermore, Inet doesn't handle Wide Area Networks (WAN) often found in large enterprises and doesn't produce random or homogeneous graphs, which are important for the study of the spread of viral code.

Closest to our work is the BRITE [36,37] system, because of the similar architectural design with NGCE. BRITE clearly separates the Graphical User Interface from the main application and allows the development of specific plug-ins in order to have other topologies covered. Additionally, BRITE provides an analysis engine to process the constructed graphs and to extract their properties. Overall, BRITE is a very effective and mature tool that efficiently covers the majority of Internet related researches. Even so, it will need some further fine-tuning to produce some popular topologies for the study of malware epidemics.

Dreier [38] presented a user-friendly graphical graph generator which builds scale-free graphs based on the Barabási and Albert model. The main shortcoming of this tool is that is not designed to handle any other topologies.

## 4 Design and Implementation

NGCE tool is written in Java. It consists of 15 Java classes, with more than 5000 lines of code besides the 150 lines of scripts.

Various scripts calculate the probability distribution function $P(k)$, which gives the probability that a node has exactly $k$ edges. After the completion of this step, $P(k)$ is automatically plotted using the gnuplot program. Every plot is displayed graphically, but is also stored in the Encapsulated PostScript (EPS) format. In addition, the title of each plot is created dynamically, in the form of an opaque Uniform Resource Identifier (URI) [39].

The title is adequate to provide all the necessary information of the plotted graph. Thus, every graph built and analyzed by NGCE can be easily reconstructed. Especially in the case of random graphs, besides the experimental data, the expected theoretical distribution function is plotted as well.

We believe that the most important instrument to analyze the output graphs is the extraction of their statistical properties via NGCE's embedded scripts. On the other hand, we are aware that a visual representation of the generated graphs would provide additional means to researchers to decide whether a graph meets their needs. In order to add this type of functionality to NGCE, we took advantage of the popular Pajek [40] tool, which is able to draw graphs in a variety of different 2 and 3-dimensional plots. To accomplish this task, we made NGCE's output graph files compatible with the Pajek tool.

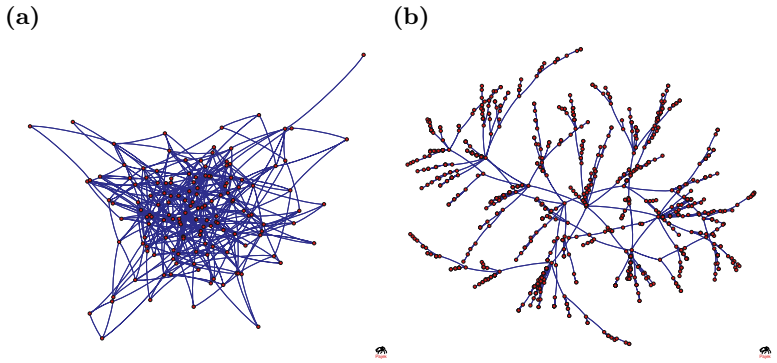


**Fig. 1.** Custom random and scale-free graphs visualized by the Pajek tool

NGCE's design is modular, based on a general-purpose Graph class that provides the main functionality for all the graph operations such as adding, removing and counting edges[1]. For each different graph topology a specific plug-in has been developed to implement the appropriate algorithm, making the development of new algorithms as separate plug-ins extremely easy.

[1] The base Graph class was based on L. Li "*Java Data Structures and Programming*" [41] book's source code used with the author's kind permission.

## 5 Analysis

In this section we analyze and comment on the graphs produced by the NGCE tool and evaluate them against the expected theoretical results.

1. **Random Graphs and Custom Random Graphs.** Following Barabási and Albert's reasoning [42], we expect the distribution connectivity of a random graph to fit well within a Poisson distribution. Indeed, numerous different random graph generation experiments confirmed our intuition, as can be see in Figure 2a. The degree distribution of the constructed graphs, in all cases, was close to the following Poisson distribution, which is valid for sufficiently large $N$.

$$P(k) = e^{-pN} \frac{(pN)^k}{k!} \tag{3}$$

2. **Scale-Free Graphs and Custom Scale-Free Graphs.** A large number of experimental runs indicated that the graphs built with the preferential attachment and incremental growth behavior follow closely the BA model (Figure 3a). The connectivity distribution of these graphs follows a power law with an exponent approximately equal to 3, as it was predicted by Barabási and Albert. Note the capability of this algorithm to produce stationary scale-free graphs; on the other hand, the model which is solely based on the preferential attachment factor exhibits temporal scale-free behavior (Figure 2b). Finally, the observed results indicate that when only the incremental growth property is enabled (Figure 3b) the scale-free characteristics of the graph tend to be diminished as the number of nodes increases. Our results are in alignment with [42], which also used numerical simulations to conclude that this model exhibits power-law scaling in its early stages where $N$ is close to $T$.

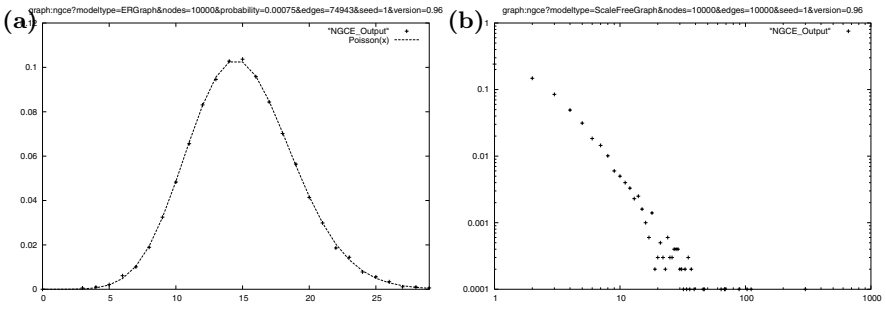


**Fig. 2.** Random graph based on the Erdős-Rény algorithm, 10000 nodes (left) and scale free graph with preferential connectivity only, 10000 nodes (right)

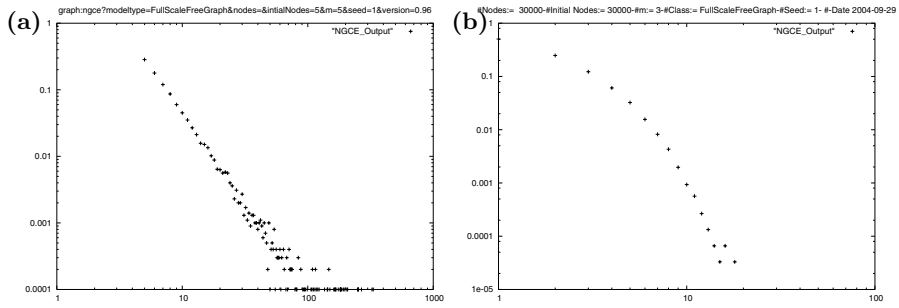


**Fig. 3.** Scale free graph with incremental growth and preferential attachment, 10000 nodes (left) and scale-free graph with incremental growth only, 30000 nodes (right)

## 6 Applications in Epidemiological Research

Our primary target for the NGCE tool was its use as a graph generator suite for research groups focused on computer epidemiology.

The comparison of the results between recent and older simulation-based studies can be a source of significant insight. The main drawback of this approach is the required effort to build identical or similar graphs in order to perform meaningful comparisons. Especially if an older study is based on different graph topologies, then the necessary additional work becomes prohibitive. As a result, very few research papers contain detailed comparisons between different algorithms on different graph topologies. NGCE's goal is to fill this gap in an efficient and effective way. We selected four of the most cited papers in the area of the computer epidemiology and reconstructed with NGCE, as close as we could, various graphs of their experiments. The outcome of this effort has been made available in the NGCE's web site for further evaluation and inspection.

We used NGCE to create most of the graph topologies which were found in Kephart's remarkable paper "How Topology Affects Population Dynamics" [8] with minimal effort. Building the homogeneous graph required only a few clicks in the NGCE's GUI or editing 3 lines in the configuration file. On the other hand, constructing Kephart's random graphs was a considerably more difficult problem, because the applied algorithm was not named or described. The author did provide, however, the statistical properties of the constructed graphs. By taking advantage of the graph analysis functions of our tool we were able to produce similar graphs. In the future, researchers can avoid such pitfalls by providing the NGCE's graph URI in their work.

Another well cited paper is "On Computer Viral Infection and the Effect of Immunization" [27] . The author describes the testbed simulation environment as a clustered topology which is typical of many transportation and energy-control networks. The above cases are among the most characteristic examples of scale-free networks, which lead us to build a scale-free graph of equal size.

Pastor-Satorras and Vespignari presented their seminal paper "Epidemic spreading in scale-free networks" [28], in which they used the Barabási-Albert algorithm to construct scale-free graphs for their experiments. We built a comparable graph using the scale-free model of our application.

Concluding this section, we note that we generated a number of graphs for own usage, as we plan to evaluate malcode containment algorithms. Specifically, we simulated the spread of a worm using the S-I [43] epidemiological model in various graphs. Where it was possible, the analytical results were evaluated against the experimental outcome of our simulator to ensure the correctness of our implementation of the algorithm and that of the underlying topology.

## 7 Conclusions

The source code of the NGCE tool as well as an extended version of this paper with additional results and figures are available at `http://istlab.dmst.aueb.gr/~vbill/ngce/` .

The NGCE tool has been designed to allow the development of the most utilized network graph topologies in an easy and reproducible way. Our aim is to assist scientists from related fields to create and use graphs without detailed knowledge of graph theory, and give them the possibility to compare their results with previous or current studies without additional effort. We are convinced that with tools, such as the NGCE, researchers can concentrate their efforts on developing efficient algorithms and understand complex system dynamics, rather than trying to build testing infrastructure.

## Acknowledgments

The work of Vasileios Vlachos is funded under the Iraklitos fellowships for research of Athens University of Economics and Business program and the European Social Fund.

## References

1. Paxson, V., Floyd, S.: Why we don't know how to simulate the internet. In: Proceedings of the Winder Communication Conference. (1997)
2. Palmer, C., Steffan, J.: Generating networks topologies that obey power laws. In: Proceedings of the GLOBECOM 2003, San Francisco, USA (2000)
3. Tangmunarunkit, H., Govindan, R., Jamin, S., Shenker, S., Willinger, W.: Network topology generators: Degree-based vs. structural. In: Proceedings of ACM SIGCOMM '02, Pittsburgh, Pennsylvania, USA (2002) 147–159
4. Dick, R., Rhodes, D., Wolf, W.: Tgff: Task graphs for free. In: Proceedings of International Workshop on Hardware/Software Codesign (Codes/CACHE '97). (1998) 97–101
5. Berk, V., Bakos, G., Morris, R.: Designing a framework for active worm detection on global networks. In: Proceedings of the IEEE International Workshop on Information Assurance, Darmstad, Germany (2003)

6. Kephart, J., Chess, D., White, S.: Computers and epidemiology. IEEE Spectrum **30** (1993)
7. Kephart, J., White, S.: Measuring and modeling computer virus prevalence. In: Proceedings of the 1999 IEEE Computer Society Symposium on Research in Security and Privacy, Oakland, California (1999) 2–14
8. Kephart, J.: How topology affects population dynamics. In: Proceedings of Artificial Life 3, New Mexico,USA (1992)
9. Chen, Z., Gao, L., Kwiat, K.: Modeling the spread of active worms. In: Proceedings of the IEEE Infocom. (2003) San Francisco, USA.
10. Zou, C., Gong, W., Towsley, D.: Code red worm propagation modeling and analysis. In: Proceedings of the 9th ACM Conference on Computer and Communication Security (CCS), Washington DC, USA (2002)
11. Staniford, S., Paxson, V., Weaver, N.: How to 0wn the internet in your spare time. In: Proceedings of the 11th USENIX Security Symposium. (2002)
12. Staniford, S.: Containment of scanning worms in enterprise networks. Journal of Computer Security (2003)
13. Scandariato, R., Knight, J.: An automated defense system to counter internet worms. Submitted to SRPS 2004, 23rd Symposium on Reliable Distributed Systems (2004) Florianapolis, Brazil.
14. Zou, C., Gao, L., Gong, W., Towsley, D.: Monitoring and early warning for internet worms. In: Proceedings of the 10th ACM Conference on Computer and Communication Security, Washington DC, USA (2003)
15. Leveille, J.: Epidemic spreading in technological networks. Hpl-2002-287, School of Cognitive and Computing Sciences, University of Sussex at Brighton, Bristol (2002)
16. Barabási, A., Albert, R., Jeong, H.: Scale-free characteristics of random networks: the topology of the world-wide web. Physica A **281** (1999)
17. Zou, C., Towsley, D., Gong, W.: Email virus propagation modeling and analysis. Technical report, University of Massachusetts Amherst, ECE TR-03-CSE-04 (2003)
18. Barabási, A., Albert, R., Jeong, H.: Mean-field theory for scale-free random networks. Physica A **272** (1999) 173–187
19. Virtanen, S.: Properties of nonuniform random graph models. Research Report HUT-TCS-A77, Helsinki University of Technology, Laboratory for Theoretical Computer Science (2003)
20. Barabási, A., Bonabeau, E.: Scale-free networks. Scientific American (2003) 60–69
21. Kanovsky, I., Mazor, S.: Models of web-like graphs: Integrated approach. In: Proceedings of the 7th World Multiconference on Systematics, Cybernetics and Informatics (SCI 2003), Orlando, Florida, USA (2003) 278–283
22. Adamic, L., Huberman, B.: Power-law distribution of the world wide web. Science **287** (2000)
23. Bornholdt, S., Ebel, H.: World wide web scaling exponent from simon's 1955 model. Physical Review E **64** (2001) 0345104 (R)–1 0345104–4
24. Faloutsos, M., Faloutsos, P., Faloutsos, C.: On power-law relationships of the internet topology. In: Proceedings of ACM SIGCOMM, Cambridge, MA, USA (1999) 251–262
25. Medina, A., Matta, I., Byers, J.: On the origin of power laws in internet topologies. ACM Computer Communication Review **30** (2000) 160–163
26. Ebel, H., Mielsch, L., Bornloldt, S.: Scale-free topology of e-mail networks. Physical Review **E 66** (2002)

27. Wang, C., Knight, J., Elder, M.: On computer viral infection and the effect of immunization. In: Proceedings of the 16th Annual Computer Security Applications Conference (ACSAC),New Orleans, Louisiana, USA. (2000) 246–256
28. Pastor-Satorras, R., Vespignani, A.: Epidemic spreading in scale-free networks. Physical Review Letters **86** (2001) 3200–3203
29. Cohen, R., K.Erez, ben Avraham, D., Havlin, S.: Resilience of the internet to random breakdowns. Physical Review Letters **85** (2000) 4626
30. Barabási, A., Albert, R.: Emergence of scaling in random networks. Science **286** (1999) 509–512
31. Yook, S., Jeong, H., Barabási, A.: Modeling the internet's large scale topology. In: PNAS Proceedings of the National Academy of Science. (2002) 13382–13386
32. Zegura, E., Calvert, K., Donahoo, M.: A quantitative comparison of graph-based models for internet topology. IEEE/ACM Transactions On Networking **5** (1997) 770–783
33. Zegura, E., Calvert, K., Bhattacharjee, S.: How to model an internetwork. In: Proceedings IEEE Infocom. Volume 2., San Francisco, CA (1996) 594–602
34. Eagan, K.C.A., Merugu, S., Namjoshi, A., Stasko, J., Zegura, E.: Extending and enhancing gt-itm. In: Proceedings of the ACM SIGCOMM 2003 Workshops, Karlsruhe, Germany (2003) 23–27
35. Winick, J., Jamin, S.: Inet-3.0: Internet topology generator. Technical Report UM-CSE-TR-456-02, EECS, University of Michigan (2002)
36. Medina, A., Lakhina, A., Matta, I., Byers, J.: Brite: An approach to universal topology generation. In: Proceedings of the International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunications Systems- MASCOTS '01, Cincinnati, Ohio, USA (2001)
37. Medina, A., Lakhina, A., Matta, I., Byers, J.: Brite: Universal topology generation from a user's perspective. In: Proceedings of the International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunications Systems (MASCOTS '01), Cincinnati, OH (2001)
38. Dreier, D.: Manual of Operation: Barabási Graph Generator v1.0. University of California Riverside, Department of Computer Science. (2002)
39. Berners-Lee, T., Fielding, R., Irvine, U., Masinter, L.: rfc2396. Available at (October 2004): http://www.ietf.org/rfc/rfc2396.txt (1998)
40. Batagelj, V., Mrvar, A.: Program for Analysis and Visualization of Large Networks, Ljubljana, Slovenia. (2004)
41. Li, L.: Java Data Structures and Programming. Springer-Verlag, Berlin, DE (1998)
42. Albert, R., Barabási, A.: Statistical mechanics of complex networks. Reviews of Modern Physics **74** (2002) 47–97
43. Daley, D., Gani, J.: Epidemic Modelling. Cambridge University Press, Cambridge (1999)