ANALYZING THE RESILIENCE AND EMERGENCE OF SUPERPEER NETWORKS

Bivas Mitra

ANALYZING THE RESILIENCE AND EMERGENCE OF SUPERPEER NETWORKS

A dissertation submitted to the
Indian Institute of Technology, Kharagpur
in partial fulfillment of the requirements of the degree

of

Doctor of Philosophy

by

Bivas Mitra

Under the supervision of

Dr. Niloy Ganguly, Prof. Sujoy Ghose

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

May 2010

APPROVAL OF THE VIVA-VOCE BOARD

Date: \ \ 20

Certified that the thesis entitled ***“Analyzing the Resilience and Emergence of Superpeer Networks”*** submitted by BIVAS MITRA to the Indian Institute of Technology, Kharagpur, for the award of the degree of Doctor of Philosophy has been accepted by the external examiners and that the student has successfully defended the thesis in the viva-voce examination held today.

(Member of DSC) (Member of DSC) (Member of DSC)

(Supervisor) (Supervisor)

(External Examiner) (Chairman)

CERTIFICATE

This is to certify that the thesis entitled ***"Analyzing the Resilience and Emergence of Superpeer Networks"****, submitted by Bivas Mitra to the Indian Institute of Technology, Kharagpur, for the partial fulfillment of the award of the degree of Doctor of Philosophy, is a record of bona fide research work carried out by him under our supervision and guidance.*

The thesis in our opinion, is worthy of consideration for the award of the degree of Doctor of Philosophy in accordance with the regulations of the Institute. To the best of our knowledge, the results embodied in this thesis have not been submitted to any other University or Institute for the award of any other Degree or Diploma.

Niloy Ganguly
Associate Professor
CSE, IIT Kharagpur

Sujoy Ghose
Professor
CSE, IIT Kharagpur

Date:

DECLARATION

I certify that

a. the work contained in this thesis is original and has been done by me under the guidance of my supervisors.

b. the work has not been submitted to any other Institute for any degree or diploma.

c. I have followed the guidelines provided by the Institute in preparing the thesis.

d. I have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.

e. whenever I have used materials (data, theoretical analysis, figures, and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the references. Further, I have taken permission from the copyright owners of the sources, whenever necessary.

Bivas Mitra

ACKNOWLEDGMENTS

This thesis is the outcome of co-operation, help and guidance of a lot of people. In my schooldays, whenever I went through the biography of Prof. Satyendra Nath Bose, I was amazed by the seemingly unreasonable power of mathematics that enables someone to discover the particles like 'Boson' without any physical experimentation! During that time, my father bought me a book titled "A Brief History of Time" written by Prof. Stephen Hawking. The content of the book was quite hard for me to digest, however, I still remember the excitement and curiosity that the book generated in my mind; the awe it created about theoretical physics by which one could comprehend bizarre phenomena's like Big bang, Black hole, life and birth of a star. After my high school, I got admitted to engineering course in computer science and soon, all my dreams about doing research in fundamental science faded out. Surprisingly, after joining the PhD program, I came across one of my supervisors Dr. Niloy Ganguly who introduced me to the subject complex network theory where I found the answers of my childhood. I have been able to relate the dynamics of Big bang, Black holes in universe with the theories of giant component and percolation in network!

I gratefully acknowledge Dr. Ganguly for his advice, supervision, and crucial contribution, which made him a backbone of my research and so to this thesis. His involvement with his originality has triggered and nourished my intellectual faculty that I will benefit from, for a long time to come. He extended

his unflinching encouragement and support towards me in both academic and non-academic sphere. I am also very much grateful to Prof. Sujoy Ghose whose truly scientific intuition has made him a constant oasis of ideas. His presence in my life since my M. Tech. days has inspired and enriched my growth as a student and researcher. I am heartily thankful to my supervisors Dr. Ganguly and Prof. Ghose whose encouragement, guidance and support from the initial to the final level enabled me to develop an understanding of the subject.

I would like to convey my special thanks to Dr. Fernando Peruani. In fact, Dr. Peruani is the one, who introduced me to the basics of statistical mechanics, modeling techniques and stochastic simulation fundamentals. I enjoyed intense work with him on different problems in various phases of my PhD work. In this line, would like to acknowledge Prof. Andreas Deutsch, Dr. Lutz Brusch and the team of Innovative Methods of Computing group, Technical University, Dresden, Germany. I have received a lot of encouragements as well as many valuable feedbacks from them during my visits to Dresden in September-October 2006 and in September-October 2007.

I would like to thank all my friends and colleagues of IIT Kharagpur who were at my side making the journey enjoyable and memorable. I am really honored to be a member of CNERG research group headed by Dr. Ganguly at CSE, IIT Kharagpur. My special thanks goes to Subrata Nandi, Joydeep Chandra and Abyayananda Maity for their useful discussions of many issues. I must confess that without the cerebral contributions of Abyay, Joydeep-da and Subrata-da, this thesis would not have taken this shape. I have really enjoyed to work with my close friend Saurav Kumar Dandapat and my student Animesh Srivastava. My special acknowledgement goes to Animesh Srivastava for his consistent support towards me in various aspects. I am also indebted in particular, to Saptarshi Ghosh and Joydeep Chandra for painstakingly proof-reading parts of this thesis. I would also like to record my heartfelt thanks to other members of CNERG group like Sanjib (Sandhu), Animesh (Mukherjee), Rajib (Maiti), Rishiraj (Saharoy), Sudipto (Saha). I would also like to extend my gratitude in particular to two B. Tech students, Md. Moin Afaque and Abhishek Kumar Dubey for helping me to develop the simulation platform.

I express my sincere gratitude to the faculty members and staffs of the department. I would specially like to thank Prof. A. Gupta, Prof. I. Sengupta, Prof. S.K. Ghosh, Prof. P. Mitra for their suggestions and encouragements. I sincerely acknowledge SAP Labs India and Prof S. V. Sankaran for providing me doctoral fellowship. I would like to thank AICTE for the National Doctoral Fellowship. I acknowledge Haldia Institute of Technology, Haldia for granting me leave for pursuing Ph.D. I am thankful to DST-BMBF and Indo-German collaborative project for funding my visits to Dresden, Germany as a visiting research scientist. I am also indebted to many organizations like MSR (India), ESMTB and conferences like ACM SIGCOMM, IEEE IPDPS, ECCS etc. for their financial assistance towards my foreign travels to participate in the conferences and summer schools.

I have been fortunate to come across many good friends in J. C. Bose Hall of Residence, without whom life would be bleak. Special thanks to Ganesh, Arijit, Mithun, Sekhar, Kamal, Saurav, Arko, Surajit, Tarit, Ritesh, Sukhen, Somok, Sudip alias our ‘Golmaal’ group. I would like to acknowledge Ankur Sahu and Amit Biswas to make my life pleasurable at the IIT Kharagpur campus. In addition, I am really thankful to Dr. Anjan Chakravorty, Anindya Banerjee and Frau Sabine Vollheim who provided me warm welcome and great hospitality during my visits in Dresden, Germany. In this scope, I would also like to thank Vinay, Manavendra, Ajit, Kausik and my others friends of Dresden for their nice company.

I am thankful to my mother for her constant support for my research and taking care of my health and food. Finally, I am really grateful to my father, who has been my personal secretary during this period managing my bank accounts, insurance savings, visa formalities, medical treatment etc. so that I can concentrate well in my research. Thank you ‘Baba’ for all the help.

Bivas Mitra
IIT Kharagpur, India

ABSTRACT

Superpeer networks are formed and maintained as a result of several node and link dynamics like bootstrapping, peer churn, attack, link rewiring etc. Significant amount of work has been done by the p2p research community in the development of efficient bootstrapping protocols. However, it is not obvious why bootstrapping of nodes and different local dynamics lead to the emergence of bimodal superpeer networks. Stability of superpeer networks also suffers from high rate of peer churn and attacks. The movements of the peers often partition the network into smaller fragments which results in breakdown of communication among peers. Although several attacks and defence techniques are discussed in the literature, less attention has been paid to assess the impact of such attacks upon the overall topology of the superpeer network. Hence, apart from the simulation and experimental study, there is a need for understanding the emergence and resilience of superpeer networks from a theoretical perspective.

In this thesis, we propose theoretical frameworks to analyze the resilience and emergence of superpeer networks against several node and link dynamics. In resilience analysis, we model the network topology and peer dynamics with the help of probability distributions and derive a critical condition for the stability of superpeer networks. The results obtained from the theoretical analysis are validated through simulation. We simulate attacks and failures on real world commercial p2p networks namely Gnutella as well as on the superpeer networks generated using theoretical degree distribution. The influence of network size as well as degree-degree correlation present in the real world networks (Gnutella) are also analyzed.

In order to understand the emergence of superpeer networks, we model bootstrapping protocol through a node attachment rule, where the probability of joining of an incoming peer to an online node is proportional to the node property (shared resource, processing power, bandwidth) and degree of the online node. We develop a formalism that calculates the degree distribution of emerging superpeer networks based upon such bootstrapping process and bandwidth constraint. We further refine the above growth framework and include dynamics like (a) peer churn and (b) link rewiring along with the bootstrapping process. The analytical framework calculates the threshold churn rate, required to break down the superpeer structure. It also discovers that in presence of proper rewiring, the QoS of p2p network shows graceful degradation in face of churn. Our theoretical model provides some empirical estimation of churn and rewiring rate of the Gnutella network which is consistent with the measurement studies.

In summary, the network resilience and other topological properties like diameter, amount of superpeers in the network, size of the largest connected component etc. play the key role on the performance of the evolving superpeer networks. We believe that proper analytical understanding will help network engineers in regulating these topological properties and subsequently improve the performance of various p2p services.

Keywords: Superpeer network, network resilience, peer dynamics, complex networks, degree distribution, giant component, generating function, network growth, bootstrapping protocols, preferential attachment.

Contents

Author's Biography

Bivas Mitra received his B.Tech. from Haldia Institute of Technology, Vidyasagar University in 2001, and M.Tech. from IIT Kharagpur in 2003 both in Computer Science and Engineering. From February 2003 to January 2006, he worked as a lecturer in the department of Computer Science and Engineering at Haldia Institute of Technology. He also worked at Soffront Software (India) Pvt. Ltd. as a Software Engineer in 2001. In January 2006, he joined as a research scholar in the department of Computer Science and Engineering, IIT Kharagpur. In his PhD tenure, he has received various fellowships like national doctoral fellowship, SAP Labs India doctoral fellowship etc. and several student travel grants to participate in different international conferences. His research interests include peer-to-peer networks, complex networks, networks modeling, optical networks, wireless internet etc.

Publications made out of this thesis (listed in reverse chronological order)

1. Bivas Mitra, Sujoy Ghose, Niloy Ganguly, "Brief Announcement: Superpeer Formation Amidst Churn and Rewiring", *ACM PODC 2010*, Zurich, Switzerland, July 2010.

2. Bivas Mitra, Abhishek Kumar Dubey, Sujoy Ghose, Niloy Ganguly, "How do Superpeer Networks Emerge?", *IEEE INFOCOM 2010*, San Diego, USA, March 2010.

3. Bivas Mitra, Abhishek Kumar Dubey, Sujoy Ghose, Niloy Ganguly, "Formal Understanding of the Emergence of Superpeer Networks: A Complex Network Approach", *ICDCN 2010*, Kolkata, January 2010.

4. Bivas Mitra, Niloy Ganguly, "Understanding the Emergence of Stable Superpeer Networks", TCPP-PhD Forum, *IEEE IPDPS*, Rome, Italy, May 2009 (poster).

5. Bivas Mitra, Niloy Ganguly, Sujoy Ghose and Fernando Peruani., "Generalized Theory for Node Disruption in Finite Size Complex Networks", *Physical Review E*, 78, 2008.

6. Bivas Mitra, Niloy Ganguly, Sujoy Ghose and Fernando Peruani, "Stability Analysis of Peer-to-Peer Networks Against Churn", *Pramana : Journal of Physics*, Springer, 71, 2008.

7. Bivas Mitra, Fernando Peruani, Sujoy Ghose and Niloy Ganguly. "Analyzing the Vulnerability of Superpeer Networks Against Attack", *ACM CCS*, Alexandria, USA, 2007.

8. Bivas Mitra, Sujoy Ghose and Niloy Ganguly, "How Stable are Large Superpeer Networks Against Attack?" *IEEE P2P*, Galway, Ireland, Sep, 2007.

9. Bivas Mitra, Fernando Peruani, Sujoy Ghose and Niloy Ganguly, "Measuring Robustness of Superpeer Topologies", *ACM PODC* 2007, Portland, USA (Brief Announcement).

10. Bivas Mitra, Sujoy Ghose and Niloy Ganguly, "Effect of Dynamicity on Peer to Peer Networks", *HiPC 2007*, Goa, India, Dec 2007.

11. Bivas Mitra, Md. Moin Afaque, Sujoy Ghose, Niloy Ganguly, "Developing Analytical Framework to Measure Robustness of Peer to Peer Networks", *ICDCN 2006*, Guwahati, India, Dec 2006.

12. Bivas Mitra, Md. M. Afaque, Niloy Ganguly. "Developing Analytical Framework to Measure Stability of P2P Networks", Poster paper, *ACM SIGCOMM* 2006, Pisa, Italy, September 2006 (poster).

List of Figures

List of Symbols and Abbreviation

α	–	Power law exponent
γ	–	Attack exponent
γ_c	–	Critical attack exponent
κ	–	Ratio of the second and first moment of the degree distribution
k	–	Degree of a node
p_k	–	Probability that a randomly chosen node has degree k
f_k	–	Probability of removal of a node of degree k
q_k	–	Probability that a node of degree k survives after node removal process ($q_k = 1 - f_k$)
f_c	–	Percolation threshold
f_r	–	Percolation threshold for random failure
f_d	–	Percolation threshold for degree dependent failure
f_{tar}	–	Percolation threshold for deterministic attack
f_p	–	Fraction of peers removed at percolation point
f_{sp}	–	Fraction of superpeers removed at percolation point
ϕ	–	Probability that a node in surviving set S will lose one link due to node removal
w_i	–	Weight of a node i
m	–	Joining node degree
f_{w_i}	–	Probability that a node joining with weight w_i
k_c	–	Cutoff degree of a node
p_{k_c}	–	Fraction of superpeer nodes
$q_{k_c(j)}$	–	Fraction of nodes joining with cutoff degree $k_c(j)$
$k_c(min)$	–	Minimum cutoff degree
$k_c(max)$	–	Maximum cutoff degree
q	–	Node joining probability
w	–	Rewiring probability
$\delta_k^{jo}, \delta_k^{rm}, \delta_k^{relink}$	–	Amount of increase in the k degree nodes due to the joining, removal of nodes and rewiring of links respectively

Pr_C	–	Peer contribution
C_{γ_c}	–	Normalizing constant
m_{ij}	–	Fraction of edges connecting the nodes of degree i and degree j
k_l	–	Peer degree in bimodal network
k_m	–	Superpeer degree in bimodal network
$\langle k_p \rangle$	–	Mean peer degree in mixed poisson network
$\langle k_{sp} \rangle$	–	Mean superpeer degree in mixed poisson network
$\langle k \rangle, z$	–	Average degree of the network
Gnutella A	–	Gnutella network simulated following the bootstrapping protocols
Gnutella B	–	Gnutella network simulated from the real world data
$CS_t(s)$	–	Component size distribution
$G(x)$	–	Generating function which generates the degree distribution p_k
N	–	Number of nodes in the network
S	–	Set of survived nodes after attack
R	–	Set of nodes removed after attack
E	–	Number of edges running between the survived set S and removed set R
M	–	Degree-degree correlation matrix

Chapter 1

Introduction

Peer-to-peer (p2p) paradigm for building distributed systems is becoming extremely popular as more and more novel applications (like VoIP, Instant Messaging, file sharing etc) are invented and successfully deployed [69]. Peer-to-peer system provides an architectural paradigm where every node performs both the role of server and client [27,58]. They exchange information and services directly with each other without any hierarchical organization or centralized control. The main advantage of the p2p paradigm is that it allows the construction of systems of unprecedented size and robustness since all clients provide resources, including bandwidth, storage space, and computing power. Thus, as nodes arrive and demand on the system increases, the total capacity of the system also increases simultaneously. This is not true for a traditional client-server architecture, in which adding more clients could mean slower data transfer for all users. Because of these desirable qualities, many researchers have focused on understanding the issues surrounding the p2p networks and improving their performance.

Peers in the peer-to-peer networks are typically connected via ad hoc overlay connections. If a participating peer knows the location of another peer in the network, then a logical link may be established from the former node to the latter. The logical links among the peer nodes form the overlay network over the physical topology. The nature of connection of this overlay network determines whether p2p system is centralized, structured, or purely decentralized. Each such class of p2p systems has its own pros and cons.

The biggest advantage of pure decentralized p2p system (where peers randomly connect each other in a self organizing manner) is its robustness. However functions upon such system tend to be inefficient; for example, search in pure p2p networks amounts to flooding the network with query messages. The flooding mechanism generates large number of redundant query packets in the network which misutilizes the valuable bandwidth and makes the unstructured P2P systems being far from scalable. Superpeer network has proved to be the solution to such problem as it can combine the efficiency of client-server architecture with the autonomy, load balancing provided by the pure p2p networks. Hence, superpeer networks have emerged as the most dominant topology among the unstructured p2p networks. Most of the commercial systems like KaZaA, Skype use superpeer networks as the underlying architecture. In these systems, nodes are selected as superpeers on the basis of their larger capacity and greater capabilities from among the set of peers. Superpeer nodes containing higher bandwidth (hence connectivity) and resource connect to each other forming the upper level in the network hierarchy. Each superpeer works as a server on behalf of a set of client peers who form the lower level of network hierarchy [139, 170]. Superpeer nodes route messages over the upper level of overlay network, and submit and answer queries on behalf of the pure peers and themselves. Hence, most of the query traffic flows through the superpeer layer (upper level) which in effect reduces the bandwidth consumption of the overall networks.

1.1 Formation and Dynamics of superpeer networks

The superpeer networks are formed and maintained as a result of several node and link dynamics like bootstrapping, peer churn, attack, link rewiring, upgradation of the peers to the superpeers etc. All these dynamics have a significant impact on the network topology as well as on the QoS of different p2p services like efficient search, file downloading etc. A brief description of these dynamics is given below. The detailed survey appears in Chapter 2.

Bootstrapping: The superpeer networks like Gnutella are formed mainly as a result of the bootstrapping protocols executed by peer servents like Limewire, Mutella, Gtk-Gnutella and Gnucleus. To join the network, incoming peers execute bootstrapping protocol through peer servents in which they discover other on-line peers in the network and send connection requests to them [34]. Bootstrapping protocols exploit physical properties of the online peers like resource content, processing power, storage space, connectivity etc. The protocols also take the finiteness of bandwidth of each online peer into consideration. At the time of joining, the incoming peer gets the list of online peers from web cache servers which are the distributed repositories for maintaining the information of 'good' online peers in the network [82]. These initial neighbor peers determine the new peer's location in the overall topology, and consequently its search and download performance. Hence, peers try (prefer) to join to 'good' (resourceful) nodes; all existing bootstrapping protocols are essentially directed towards fulfilling this basic objective [62, 104].

Peer churn and attacks: In superpeer networks, a peer joins the system when a user starts the servent, uses available resources of other peers (e.g., CPU, storage, bandwidth) while offering its own resources, and leaves the system when the user exits the application at some arbitrary later point in time. The independent arrival and departure by thousands or millions of peers create the collective effect of peer churn. Churn significantly affects both the design and evaluation of P2P systems, overlay structure [159] and the resiliency of the overlay [158]. In addition to that, important peers are also targeted for attack [138, 145]. Denial Of Service (DoS) attack [138] drown important peers in fastidious computation so that they fail to provide any service requested by other peers. Attackers mount more powerful attacks by leveraging the resources of multiple peers; these attacks are known as distributed denial of service (DDoS) attack [136]. Eclipse attack, Sybil attack, worm propagation, file poisoning, file pollution [26,145] are some of the important attacks that also affect the connectivity of the p2p networks. In summary, these peer churn and attacks cause serious threat to network resilience[1] as they have the potential of breaking down the connectivity among the peers in the network.

Link rewiring: Link rewiring is another internal dynamics that frequently occurs within the p2p networks. The peer node often disconnects the existing connection

[1] In this thesis, we do not differentiate between the terms stability and resilience. They are therefore used interchangeably.

with its neighboring nodes and establishes new connections with other 'good' online peers in the network. This rewiring operations, executed by peer servent, improve QoS of different p2p services by keeping 'resourceful' nodes as neighbors and play a major role in maintaining connectivity among the peer nodes specially in the challenged environment of churn and attack.

In summary, peer dynamics (churn and attack) disrupt the connectivity among the nodes, hence affect the stability of superpeer networks. In addition, bootstrapping and link rewiring play a major role in the formation and maintenance of the superpeer topology. The performance of network is affected by several topological parameters like amount of superpeer nodes, network connectivity, diameter etc. Hence, proper understanding of all these dynamics and their influence on various topological parameters will help in shaping the QoS of different p2p services.

1.2 Challenges in p2p networks and limitations of the classical approach

Significant amount of work has been done by the p2p research community in the development of efficient bootstrapping protocols [31, 62, 78, 130, 156, 169]. Several of these protocols assume the presence of a centralized bootstrap server [81] later modified to function just in presence of distributed GWebCache [34, 82, 146]. In addition, several random address probe based [31,61] and locality aware bootstrapping protocols [34] are developed to minimize the bootstrapping time and to reduce the redundant traffic in the underlay topology. Most of the works done in this field are directed towards designing of bootstrapping protocols to improve the performance of the p2p services. Such ad hoc improvements seem to have limited utility compared to the overhead they incur. Side by side, it is not obvious why bootstrapping of nodes and other local dynamics lead to the emergence of bimodal network, which appear in superpeer networks like Gnutella. It is interesting to note that these activities (joining, churn, rewiring) are completely driven by individual peer servents who perform them to optimize their own quality of service oblivious of the performance of the entire network. Hence, there is no explicit effort by either Gnutella client or server program towards formation of superpeer topology. The performance of the superpeer networks heavily depends upon the topological properties of the emerging networks [20,139,144, 170]. This includes the network diameter, amount of superpeers in the network, peer-superpeer ratio etc. Hence, regulating these topological properties and subsequently improving the performance of various p2p services will prove to be an *useful* step for p2p research community. Due to its decentralized nature of formation, controlling the topological structure of the superpeer network is not a trivial task. However, to the best of our knowledge, little work has been done to calculate these network parameters that emerge through the bootstrapping, churn and rewiring processes. Hence a considerable amount of research need to be directed towards modeling node and link dynamics and analytically understanding their exact impact on the topology of the emerging superpeer networks. Proper understanding of the various node and link dynamics and their impact on the emergence of superpeer networks may provide some important insights to the network engineers for improving the quality of various p2p services.

Peer-to-peer networks also suffer from high rate of peer churn due to the continuous joining and leaving of the nodes in the network. In addition, stability of the network can get affected through intentional attacks targeted towards important peers [138,145]. These dynamics of the peers often partition the network into smaller fragments which result in breakdown of communication among peers. We find that although several attacks and defence techniques are discussed in the literature, less attention have been paid to analyze the impact of such attacks upon the overall topology of the p2p networks. It is very important to maintain the connectivity of p2p network in order to sustain the regular p2p activities. Measurement based study and experimental analysis of resilience of p2p networks have been done by various researchers [146,158,159]. Some simulation based studies have also proposed design guidelines to construct robust p2p networks [130]. However, apart from simulation and experiment based study, stability analysis of the peer-to-peer networks also need to be undertaken from a theoretical perspective. More specifically for superpeer networks, design engineers often face the essential questions like, what is a good ratio of peers to super-peers in the network? How should superpeers connect to each other and with regular peers? How different topological parameters may affect the network stability against various node dynamics? How does size of the network play a role in determining stability? In summary, we can say that a comprehensive theory for understanding the stability of finite sized networks under any type of node disturbance is desirable. In this thesis, we try to address some aspects of these issues related to the stability and emergence of superpeer networks.

1.2.1 Complex network as a toolbox

The commercial peer-to-peer networks are quite large in size and contain millions of nodes. As discussed in the previous section, these large scale p2p networks are formed and maintained as a result of various node and link dynamics. P2p networks at any instant of time can be viewed as very large scale dynamic graph. However, it is difficult to apply traditional graph theoretic approaches for analyzing the properties of such large scale networks which are in a constant state of flux. Hence, the behavior of these systems can only be analyzed by observing various statistical properties of the network especially by applying the theories of network science. Large scale dynamic networks are found in various fields of study like sociology (social network, friendship network, film actors network), biology (protein-protein interaction network, metabolic network), linguistics (word co-occurrence network), information science (citation network), electrical technology (power grid, electronic circuits), computer science (internet, world wide web network) etc. Network theoretic approaches are widely used in analyzing these social, biological, and technological networks which display non-trivial topological features like heavy tail in the degree distribution, a high clustering coefficient, assortativity or disassortativity among vertices, community structure, and hierarchical structure. Large scale p2p networks can also be modeled as complex graphs and various theories related to network science may be applied in analyzing the behavior of p2p networks. Significant amount of work has been done in the field of complex networks in understanding the growth of complex networks in face of various node dynamics [11, 16, 18, 45, 87]. The basic assumption of all these works has been that a node joins the network based on preferential attachment, that is a new node generally attach itself to 'important' existing nodes. It has been widely seen that such behavior leads to the emergence of scale free or power law networks. The theoretical analysis of the formation and dissolution of giant component against random failures and attacks in large scale networks are mostly based upon the percolation theory and are discussed in [9, 28, 29, 127].

In this thesis, we utilize various concepts of complex network theory like percolation theory, continuum theory, etc and suitably modify them to analyze the dynamics of superpeer networks. The main contribution of the thesis is two folds;

1. Analyzing the stability of arbitrary size superpeer networks in face of peer churn and attacks,
2. Formal understanding of the emergence of superpeer networks in face of various local events like churn, link rewiring etc.

1.2.2 Objectives of the thesis

The principal objective of the thesis is to develop analytical frameworks for understanding the various dynamics in large scale dynamic superpeer networks. We primarily focus on two major topological properties, namely resilience and emergence of superpeer networks. Specific problems are :

- Development of an analytical framework to measure network stability against node dynamics like peer churn and attack.

  One of the main objectives of this thesis is to build up a complete analytical framework whereby given a topology and an attack scenario, one should be able to predict the exact point of breakdown of the network. Such a framework should also be able to explain the observed topological characteristics of superpeer networks. The effect of peer-superpeer degrees (and their respective fractions) on the network stability need to be illustrated. The network gets deformed after removal of a fraction of nodes along with their adjacent links due to node dynamics. The developed framework need to also precisely describe the topology of the deformed network after churn or attacks.

- Modeling bootstrapping protocols as node attachment rules and formally explaining the emergence of superpeer networks.

  Node attachment rules may be influenced by factors like shared resources, processing power, bandwidth etc. These abstract parameters need to be quantitatively represented and their impact on the topological properties of the superpeer networks need to be analyzed through such a growth framework. The framework should also be able to illustrate the impact of peer churn and rewiring on the network properties like amount of superpeer nodes, network connectivity, component sizes, network diameter etc. All these parameters have significant influence upon the quality of different p2p services. Hence the final objective of this thesis is to build up a comprehensive framework encompassing growth, node churn and link rewiring.

Keeping these above broad objectives in mind, the particular work done is outlined in the next section.

1.3 Contribution of the thesis

In this thesis, we develop theoretical frameworks to analyze the resilience and emergence of superpeer networks against several node and link dynamics. These frameworks are validated through simulation as well as real world data of Gnutella. We also discover several nonintuitive, interesting properties related to network topology that may be useful to the network researchers to improve the performance of the p2p services. The specific contributions are given below.

- **Stability analysis against peer churn and attacks**
  We model the superpeer networks with the help of degree distribution p_k (probability of a node of degree k) and peer dynamics by another probability distribution f_k (probability of removal of a node of degree k). We derive a critical condition for the stability of superpeer networks which undergo node dynamics. The degree distribution of the deformed network after node removal is

also calculated. The results obtained from the theoretical analysis are validated through stochastic simulation as well as by real data of Gnutella network. We measure the impact of fraction of superpeers in the network as well as their connectivity upon the stability of the network. The influence of network size as well as degree-degree correlation present in the real world networks like Gnutella is also analyzed.

- **Generalized model of node dynamics**
  We characterize the node dynamics as the various kinds of node removal processes. We view **degree dependent attack** as a broad class of node removal process which is able to capture peer churn and attacks. In this node removal strategy, the probability of removal of a node (f_k) having degree k is proportional to k^γ. We show that, by varying the attack parameter γ, we can generate the wide range of node dynamics, from random failure to deterministic attack.

- **Emergence of superpeer networks due to bootstrapping**
  We model bootstrapping protocols through node attachment rules. We show that a significant class of bootstrapping protocols may be viewed as a node attachment rule where the probability of joining of an incoming peer to an online node is proportional to the node property (shared resource, processing power, bandwidth) and degree of the online node. We identify that in p2p networks, bandwidth of a node is finite which restricts its maximum degree. A node, after reaching its maximum degree, rejects any further connection requests from incoming peers. We develop a formalism that calculates the degree distribution of an emerging superpeer networks based upon such bootstrapping process and bandwidth constraint. The proposed growth framework reveals that the interplay of finite bandwidth with node property plays a key role in the accumulation of superpeer nodes in the network. As an application study, we show that our framework, with some modification, can explain the topological configuration of commercial Gnutella networks.

- **Emergence of superpeer networks against peer churn and link rewiring**
  We refine the above growth framework where emergence of the superpeer networks is driven by the (a) joining of incoming nodes (b) random departure of peers due to peer churn and (c) rewiring of the existing links thereby biasing

connections towards resourceful peers. The analytical framework calculates a critical churn rate, upto which the qualitative nature of superpeers is preserved. It also discovers that in presence of proper rewiring, the QoS of p2p network shows graceful degradation in face of churn. Our theoretical model provides some empirical estimation of the node properties, churn and rewiring rate of the Gnutella network which is consistent with the measurement results.

1.4 Organization of the thesis

The organization of rest of the thesis is as follows. Prior to dealing with the proposed work, we report a survey on related research topics in Chapter 2. Chapter 3 focuses on modeling of superpeer networks as well as various node dynamics and analyzes the stability of superpeer networks against peer churn. In Chapter 4, we analyze network stability and topological deformation against attacks. Chapter 5 develops a formal framework to analyze the emergence of superpeer network from bootstrapping by incoming peers. In Chapter 6, we extend the framework to include peer churn and link rewiring. Chapter 7 concludes the thesis.

Chapter 2

Literature survey

This thesis builds up an analytical framework to understand the resilience of superpeer networks against various types of node failures. It also reports a comprehensive analysis explaining the emergence of superpeer networks. In this context, this chapter provides an upto date survey of the various works done in the field of network stability and growth of networks. The organization of the survey is as follows; the first section covers different kinds of peer-to-peer networks with special emphasis given on superpeer networks. Second section of the survey reports different disruptive events in p2p networks like peer churn, attacks and their defence strategies. Further we review the stability analysis of the large scale networks in the perspective of complex network theory. The third section discusses the formation of p2p networks through bootstrapping and other local events like link rewiring. In this perspective, we review the various theories related to the growth of complex networks.

2.1 Introduction to peer-to-peer networks

Peer-to-peer networks belong to the paradigm of computer networks where each workstation has equivalent capabilities and responsibilities [27, 100]. The main advantage of the p2p networks is that it allows the construction of systems of unprecedented size and robustness since all clients provide resources, including bandwidth, storage space, and computing power. Peers in the p2p networks establish an application layer connectivity among themselves, which is known as overlay. If a participating peer knows the address of another peer in the network, then a link may be created from the former node to the latter in the overlay network. Based on how the nodes in the overlay network are linked to each other, the current p2p architecture can be classified into three types [98, 166], centralized, decentralized and structured, decentralized but unstructured.

1. Centralized : In centralized system, a list of index items corresponding to the shared files in the network is kept in a centralized server in the form of ⟨*object-key, node-address*⟩ table. Each arriving node needs to actively notify the centralized server about the files (objects) it possesses. Therefore the querying node needs to contact the central server to obtain the peer's addresses containing its searched object. However, at the time of downloading the searched object from the peer, the querying node directly establishes the connection with the concerned peer and download the item. This type of p2p architecture is very simple, efficient and easily deployable. But like any centralized system, it has the problem of single point of failure and lacks scalability. The most popular example of centralized p2p system is Napster [4]. After it's inception in May 1999, many record companies realized that the threat Napster posed to its potential earnings was immense and hence need to be legally challenged. This court case involved the Recording Industry Association of America (RIAA), which includes such music industry giants as AOL Time Warner's Warner Music, BMG, EMI and Sony Music among others, suing Napster over breach of copyright law. This forced Napster to shut down the file-sharing service of digital music, which was literally its *killer application*.

2. Decentralized and structured : Structured p2p network employs a globally consistent protocol to ensure that any node can efficiently route a search query to a peer that has the desired file. Such structured p2p systems use Distributed Hash Table (DHT) as a substrate, in which data object (or value) location information is placed deterministically at the peers with identifiers, corresponding to the data objects unique key [24,98]. DHT-based systems have a property that consistently assign uniform random NodeIDs to the set of peers into a large space of identifiers. Data objects are assigned unique identifiers called keys, chosen from the same identifier space.

Keys are mapped by the overlay network protocol to a unique peer in the overlay network. The p2p overlay networks support scalable storage and retrieval of {key, value} pairs on the overlay network. Given a key, a store operation (put(key,value)) and retrieval operation (value=get(key)) can be invoked respectively to store and retrieve the data object corresponding to the key, which involves routing requests to the peer corresponding to the key. Each peer maintains a small routing table consisting of its neighboring peers NodeIDs and IP addresses. Lookup queries or message routing are forwarded across overlay paths to peers in a progressive manner, with the NodeIDs that are closer to the key in the identifier space. Different DHT-based systems have different organization schemes for the data objects and its key space and routing strategies. In theory, DHT-based systems can guarantee that any data object can be located in a small $O(\log N)$ overlay hops on average, where N is the number of peers in the system. Since structured overlays impose rigid topologies on the participating nodes, they are often very limited in their ability to adapt to the sudden departure of nodes [92, 141]. This leads to comparatively high round trip times within the overlay and unnecessarily increases the load imposed on the underlay. Hence in structured p2p network, the network resilience and adaptability is compromised at the cost of search efficiency. Some well known DHTs are Chord, Pastry, Tapestry, CAN, and Tulip [98].

3. Decentralized and unstructured : An unstructured p2p system is composed of peers joining the network using some defined rules, without any prior knowledge of the topology. As no special network structure needs to be maintained, unstructured p2p systems are extremely resilient to peer churn. In this category, the overlay networks organize peers as random graph in flat or hierarchical manner (e.g. Super-Peers layer) and use techniques like flooding, random walks or expanding-ring Time-To-Live (TTL) search etc on the graph to query content stored by overlay peers [6]. Searching in unstructured networks is often based on flooding or its variation because there is no control over data storage [100]; data are stored among peers without any specific rule. During flooding, nodes send query messages across the overlay with a limited lifetime. When a peer receives the flood query, it sends a list of all contents matching the query to the originating peer. Since there is no correlation between a peer and the content managed by it, there is no guarantee that flooding will find a peer that has the desired data. However, due to the high dynamicity of peers, robustness is given the topmost priority. Most of the popular p2p networks such as Gnutella and FastTrack are unstructured [65] in nature.

2.1.1 Limitations of unstructured systems: motivation behind superpeer networks

Although unstructured p2p systems have many strengths, it also suffers from certain serious limitations. **(a)** As previously mentioned, search in pure unstructured p2p networks amounts to flooding the network with query messages. In this technique, query packets are propagated to all neighbors within a certain radius until the desired object is found. However, this flooding mechanism generates large number of redundant query packets in the network which wastes the precious bandwidth and makes the unstructured p2p systems being far from scalable. **(b)** In addition, the search queries may not always be resolved in unstructured p2p networks. Popular content is likely to be available at several peers but if a peer is looking for rare data shared by only a few other peers, then it is highly unlikely that search will be successful [41]. **(c)** Another important source of inefficiency is bottlenecks caused by the very limited capabilities of some peers. In pure p2p networks, all peers are given equal roles and responsibilities, regardless of their capabilities. One study [163] found that peers connected by dialup modems become saturated by the increased load; this leads to a huge departure of these kind of peers, resulting in network fragmentation. Moreover, studies such as [146] have shown considerable heterogeneity (e.g., up to 3 orders of magnitude difference in bandwidth) among the capabilities of participating peers. These insights lead to the appearance of superpeer network, which is detailed next.

Superpeer network:

One of the obvious ways of eliminating the limitations of pure p2p systems is to take advantage of node heterogeneity, hence assigning greater responsibilities to high bandwidth, resourceful nodes namely superpeers. Superpeer nodes are selected for their larger capacity and greater stability from among the set of peers. In superpeer networks, superpeer nodes connect with each other forming the upper level in the network hierarchy. Each superpeer works as a server on behalf of the set of pure or regular peers who form the lower level of network hierarchy [103, 139, 170]. Superpeer nodes route messages over the upper level of overlay network, and submit and answer queries on behalf of the pure peers and themselves. Since superpeers act as centralized servers to the pure peers, they can handle queries more efficiently than each individual peer could. However, since there are relatively many superpeers in a system, no single superpeer has to handle a very large load, nor will one peer become a bottleneck or single point of failure for the entire system. Hence superpeer networks have the potential to match the performance and scalability of structured systems, while retaining the benefits of unstructured p2p systems [76, 103, 139]. In the following section, we illustrate the basic architecture of superpeer networks and report a brief review on various kinds of superpeer networks proposed in the literature.

2.1.2 Superpeer networks design

Several protocols and design methodologies have been proposed to optimize superpeer network topologies [59, 60, 85, 139, 170]. Initially, major thrust was given on the construction of the robust network with proper load balancing and search efficiency [117, 139, 170]. These works showed that selection of the superpeer nodes, transformation of peers to superpeers, and the association of peers with some chosen superpeers play a major role in the network performance. Yao-graph based topologies also come as an alternative as these approaches offer simple construction rules and also ensure scalability and performance [85, 99]. In addition to this, connecting superpeer and peers based upon semantic similarities [59, 60] and communicating distance [79] have proven to be useful to optimize several p2p services like search, file download latency etc. A brief review on the design of superpeer networks follows.

In [170], Molina et al. presented a set of design guidelines as summarizing the main tradeoffs in superpeer networks. The network performance is measured based on two types of metrics, (a) load, and (b) quality of results. They looked at both individual load (the load of a single node), as well as aggregate load, (the sum of the loads of all nodes in the system). The quality of results is measured by the number of results returned per query. They showed that increasing the number of superpeers in the network reduces the load of the individual superpeers. However, it increases the aggregate load in the network by increasing the overall traffic in the superpeer layer of the overlay, hence a balance need to be struck somewhere. They also discovered that superpeer redundancy has no significant effect on aggregative performance, however redundancy does decrease individual super-peer load significantly. They formulated a general procedure that incorporates these thumb rules and produces an efficient topology. Finally, they discussed how an individual node without a global view of the system might make local decisions to form a globally efficient network. In [139], Pyun et al. proposed a distributed protocol for the construction of a balanced low-diameter superpeer topology (Scalable Unstructured p2p System) at low cost. SUPS is an unstructured p2p system in which the interconnections between superpeers are selected to approximate a random graph. In the proposed network, superpeers are organized such that the resulting overlay network will have a balanced load and a logarithmic diameter, with minimum node degree. The protocol is shown to be robust to (a) rapid changes in the set of superpeers, and (b) failures in the superpeers.

Montresor et al. [117] proposed a mechanism for the construction of robust superpeer topologies based on the well-known gossip paradigm. Here each node periodically initiates an information exchange with another peer, selected randomly. Based on this information, a client may decide to become a superpeer and take responsibility for some of the clients of the other node, to alleviate its load; alternatively, a superpeer may decide to move all its clients to the other node and become a client by itself, to reduce the number of superpeers and thus the traffic generated by communication between superpeers. The continuous gossiping of topology information captures the dynamic nature of p2p systems and makes the network information consistent: nodes may learn about a new node by receiving its identifier in an exchange, while crashed nodes are progressively forgotten and then removed from the network. Furthermore, the protocol is also claimed to be efficient as the total number of messages exchanged among all nodes scale linearly with the size of the network.

Kleis et al. [85] studied the performance of Yao-Graph based superpeer topologies. Using Yao-Graphs, they achieved a global characteristic of the superpeer topology by applying simple local construction algorithm. They claimed that due to the lightweight structure of Yao-Graphs, the resulting networks have promising properties regarding scalability and performance, while still offering the benefits of the p2p approach with regard to network resiliency. In [79], Gian et al. presented a self-organizing, decentralized protocol capable of building and maintaining superpeer-based, proximity-aware overlay topologies. The goal of the protocol is to build a topology where peers and superpeers are connected based on their distance (measured by communication latency). The proposed algorithm used gossip-based protocol to spread messages to nearby nodes and biology-inspired task allocation mechanism to promote the 'best' nodes to superpeer status and to associate them to nearby peers. In a similar kind of work, Lua et al. [99] designed underlay-aware topologies connecting all nodes that offer promising properties in terms of excellent communication quality. They exploited the underlying network locality and proximity of the nodes for overlay routing and node placement strategy. In this work, Yao-graph based approach has been used to build the connectivity at superpeer layer; the resulting outcome brings that, every superpeer getting connected to six closest superpeer neighbors.

Garbacki et al. [59,60] introduced a self-organizing superpeer network architecture (SOSPNet) that reflects the semantic similarity of peers, sharing contents across the users of wide variety of interests. SOSPNet uses two-level semantic caches deployed at both the superpeer and the peer level to maintain relationships between related peers and files. The cache maintained by a superpeer contains references to those files which were recently requested by its peers, while the cache of a peer stores references to those superpeers that satisfied most of its requests. They have shown how this simple approach can be employed, not only to optimize searching, but also to solve generally difficult problems encountered in p2p architectures such as load balancing and fault tolerance.

2.1.3 Networks modeling

Superpeer topology can be represented by a complex graph structure which have evolved 'naturally' and hence it falls under the category of random graphs. In random graphs, network topologies are represented by the degree distribution p_k which signifies the probability that a randomly chosen node is of degree k. In other words, it represents the fraction of nodes in the network of degree k. There has been several attempts to represent p2p networks using theoretically well known graphs. Also measurement studies to characterize degree distribution of real world p2p network have also been made. We provide a brief sketch of both.

Erdos and Renyi graph: In the well-known Erdos and Renyi (E-R) random network [48], every pair of nodes is linked with a probability p. The degree distribution of this network follows Poisson degree distribution

$$p_k = \frac{\langle k \rangle^k e^{-\langle k \rangle}}{k!} \tag{2.1}$$

The average degree $\langle k \rangle = Np$ where N is the number of nodes. In E-R graph, p_k peaks at an average $\langle k \rangle$ and decays exponentially for large k leading to a fairly homogeneous network, in which each node has approximately the same number of links $k \simeq \langle k \rangle$.

Scale-free networks: In contrast, results on the Internet, world-wide web (www) [8] and other large networks indicate that many systems belong to a class of inhomogeneous networks, referred to as scale-free networks, for which p_k decays as a power-law, i.e. $p_k \sim k^{-\alpha} e^{k/\kappa}$ [28] where α and κ are constants. In fact, most of the real world networks do not exhibit power law behavior at large degrees k; p_k falls exponentially for high k. This is the reason behind including the exponential cutoff $e^{k/\kappa}$ in the power law degree distribution. In most cases, the power law exponent varies between $\alpha = 2.15$ to 2.3 [146]. While the probability that a node has a very large number of connections ($k >> \langle k \rangle$) is practically prohibited in exponential networks, highly connected nodes are statistically significant in scale-free networks.

Bimodal network: [133] introduced star networks of N nodes with degree distribution

$$p_k = \begin{cases} (N-1)/N; & k = 1 \\ 1/N; & k = N-1 \end{cases} \tag{2.2}$$

and $p_k = 0$ for all other values of k. Next, they extended the star network to general bimodal networks where q high degree hubs connected to the remaining nodes of degree one. For networks with average degree $\langle k \rangle$, the degree distribution is specified as

$$p_k \;=\; \begin{cases} (N-q)/N; & k=1 \\ q/N; & k=k_2 \end{cases} \tag{2.3}$$

where $k_2 = \frac{(\langle k\rangle - 1)N+q}{q}$ and $p_k = 0$ for all other k.

Gnutella network: Initial measurement studies confirmed that, degree distribution of Gnutella network, that continuously expand by the addition of new nodes through preferential attachment, follows a power-law distribution with exponent $\alpha = 2.3$ [142, 146]. Further measurement through sophisticated crawlers [143] have shown that, Gnutella degree distribution follows a multi-modal distribution, combining a power law and a quasi-constant distribution. The results show that although Gnutella is not a pure power-law network, it preserves good fault tolerance characteristics while being less dependent than a pure power-law network on highly connected nodes that are vulnerable to attack. The topological structure of Gnutella obtained from another measurement study confirms that its degree distribution does not exactly follow power law distribution [159]. Rather, accumulation of superpeer nodes [103] shows some modal behavior at the high degree nodes that gives rise to bimodal degree distribution.

To conclude, we report a very interesting study on mathematical modeling of superpeers in the field of polymer science [19]. This paper investigated how the network topology of an ensemble of telechelic polymers changes with temperature. The telechelic polymers serve as 'links' between 'nodes', which consist of aggregates of their associating end groups. They showed that the degree distribution of this system closely resembles superpeer networks and consists of two Poissonian distributions. They modeled the network as the superposition of two Poisson distributions with different average degree $\langle k\rangle$. Nodes in the distribution with higher average degree $\langle k\rangle_{SP}$ are called superpeers and others are peers with low average degree $\langle k\rangle_P$. They showed that below the 'micelle transition', the topology can be described by a robust bimodal network in which superpeer nodes are linked among themselves and all peer nodes are linked only to superpeers. At lower temperatures the peers completely disappear leaving a structure of interconnected superpeers.

2.2 Dynamics on peer-to-peer networks

In this section, we provide a comprehensive survey on various node dynamics like peer churn and attacks that occur in p2p networks. The churn and attacks can be modeled and their impact has been traditionally analyzed using complex network approach; a detail description of these approaches is described in the next subsections.

2.2.1 Churn in p2p networks

In p2p networks, a peer joins the system when a user executes the peer servent, uses available resources of other peers (e.g., CPU, storage, bandwidth) while offering up its own resources, and leaves the system when the user exits the servent at some arbitrary later point in time. One such join-participate-leave cycle may be defined as a *session*. Peers may join and leave the system at any arbitrary time. This implies that (i) peer participation in p2p systems is inherently dynamic, and (ii) these dynamics are primarily user-driven. The user-driven dynamics of peer participation, or churn, must be taken into account in both the design and evaluation of any large scale p2p application. Churn significantly affects both the design and evaluation of p2p systems, overlay structure [159], the resilience of the overlay [91], and the selection of key design parameters [92].

Researchers and developers performed several studies on churn in order to understand its implications on peer-to-peer networks. Next we review peer churn in three different perspectives (i) Measurement based studies (ii) Modeling and (iii) Strategies to minimize the effect of churn

Measurement based study: In [146] Saroiu et al. performed a measurement study of the two popular peer-to-peer file sharing systems, namely Napster and Gnutella. The paper found that Gnutella presents a highly robust overlay in the face of random breakdowns; the overlay fragments only when more than 60% of the nodes shut down. Stutzbach et al. [158] made a major contribution towards understanding churn by conducting deeper analysis and relying on more accurate measurements. They studied churn in three types of widely-deployed p2p systems: Gnutella, Kad, and BitTorrent. One of the most basic parameters of churn is the session length distribution, which captures how long peers remain in the system each time they appear. Their experimental results showed that while most sessions are short (minutes), some sessions are very long (days or weeks). The data is better described by Weibull or log-normal distributions. They also report that the distribution of session lengths does not significantly change over time. Hence, the measurement of past session length is a good predictor of the next session length. The availability of individual peers also exhibits a strong correlation across consecutive days. In [69] Guha et al. presented the measurement study of the Skype VoIP system. They observed that there is very little churn in the superpeer layer of the network. Further, they reported that session lengths are heavy-tailed and are not exponentially distributed. Hence, they concluded that the population of supernodes in the system tends to be relatively stable; thus node churn, a significant concern in other systems, seems less problematic in Skype. In [160], Stutzbach showed that the Gnutella overlay is extremely robust to random peer removals. For instance, after removing 85% of peers randomly, 90% of the remaining nodes are still connected. in this case, long-lived superpeers form a stable and densely connected core overlay (*onion-like structure*), providing stable and efficient connectivity among participating peers despite the rapid dynamics of peer participation.

Modeling: One of the first models of churn was proposed in [130], where arrival of new nodes follow Poisson distribution with rate λ, and the duration of time a node stays connected to the network is independently and exponentially distributed with parameter μ. Leonard et al. [91] examined two aspects of network resilience in dynamic p2p systems; (a) ability of each user to stay connected to the system in the presence of frequent churn and (b) partitioning behavior of the entire network. In this work, 'resilience' generally refers to the ability of an user i to stay connected to the rest of the graph for duration (lifetime) L_i while its neighbors are constantly changing. To examine the behavior of churn, this paper introduced a simple node-failure model based on user lifetimes and studied the resilience of p2p networks in which nodes stay online for random periods of time. The results indicated that systems with heavy-tailed lifetime distributions are more resilient than those with light-tailed (e.g., exponential) distributions. They further showed that k-regular graphs offer the highest local resilience among all systems with a given average degree. Yao [171] introduced a generic model of heterogeneous user churn and derived the distribution of the various metrics observed in prior experimental studies (e.g., lifetime distribution of joining users, joint distribution of session time of alive peers, and residual lifetime of a randomly selected user). In [15] Ranjita et al. studied several characteristics of host availability in the Overnet peer-to-peer file sharing system, and discussed the implications of the findings on the design and operation of peer-to-peer systems. They modeled peer availability by a combination of two time-varying distributions: (1) short-term daily enter and exit of individual hosts, and (2) long-term host arrivals and departures.

Strategies to combat churn: In [130] Pandurangan proposed neighbor replacement protocol as a result of lost connections due to peer churn. Analysis showed that the protocol results in a constant degree network that is likely to stay connected and have small diameter. Some work has been done on the design of stable distributed network in a proactive manner [66]. In these networks, churn rate is reduced by intelligently connecting the network by a selected set of joining nodes. There are two different strategies for node selection. First of all, the use of previous information about nodes to attempt to predict which nodes will be stable. These (Predictive Fixed) strategies are often used in the deployment of services on PlanetLab, where developers pick a set of machines and run their applications exclusively on those machines for days or months. The second strategy (Agnostic Replacement strategy) is to replace a failed node with a new one. The different strategies followed are (1.) Random Replacement (RR): replace a failed node with a uniform-random available node and (2.) Preference List (PL): rank the nodes according to some preference order and pick the top k available nodes. The paper also provided a comparison of the performance of a range of different node selection strategies using real-world traces.

2.2.2 Attack and defence strategies in p2p networks

Understanding the effect of attacks upon the large scale peer-to-peer networks is becoming a major challenge for the p2p network community. We report various attacks and defence techniques that are commonly employed in p2p networks.

The most prominent attack that affects the stability of the network is **Denial Of**

Service (DoS) attack [47] which is gradually becoming huge threat to the Internet community [157]. DoS attack drowns important peers in fastidious computation as a result of which they fail to provide any service requested by other peers. Alternatively, DoS attack takes an attempt to flood the network with bogus packets, thereby preventing legitimate network traffic. In addition to that, attackers mount more powerful attacks by leveraging the resources of multiple peers; these attacks are known as **distributed denial of service** (DDoS) attack [121, 122, 136]. The perpetrator in DDoS attack remotely installs the slave programs in the peers with poor security, and at the right time instructs thousands of these slave programs to attack a particular target. DDoS attacks are extremely hard to block, as a malicious user can use an enormous number and diversity of machines to launch the attack. In addition, as the attacker is often only indirectly involved, it becomes impossible to identify the source of the attack. The first problem is detecting a DoS attack as it can be mistaken with a heavy utilization of the machine. A widely used technique to hinder DoS attacks is 'pricing'. The host will submit puzzles to his clients before continuing the requested computation, thus ensuring that the clients go through an equally expensive computation. If each attempt to flood his victim results in him having to solve a puzzle beforehand, it becomes more difficult to launch a successful DoS attack.

In **Sybil attack** [46, 138], an attacker subverts the reputation system of a peer-to-peer network by creating a large number of pseudonymous entities, using them to gain a disproportionately large influence. Once the control has been accomplished, the attacker can abuse the protocol to disconnect the different parts of the network. A reputation system's vulnerability to a Sybil attack depends on how cheaply identities can be generated, the degree to which the reputation system accepts inputs from entities that do not have a chain of trust, linking them to a trusted entity, and whether the reputation system treats all entities identically. A good defense is to render a Sybil attack unattractive by making it impossible to place malicious identities in strategic positions. Another proposition could be to include the nodes IP address in its identifier as a malicious node would thus not be able to spoof fake identities. In summary, carefully configured reputation-based systems might be able to slow the attack down [46].

In **eclipse attack** [138,151], attackers gain control over a certain amount of nodes along strategic routing paths. Once this is achieved, then it is possible to inefficiently reroute each message and drop all messages he receives, thus completely separating both subnetworks. The main defense against eclipse attacks is simply to use a pure p2p network model [138]. If the nodes in a p2p network are randomly distributed, then there are no strategic positions and an attacker cannot control his nodes' positions. In [152] Singh et al. presented a defense technique against eclipse attacks based on anonymous auditing of nodes' neighbor sets. If a node has significantly more links than the average, it might be mounting an eclipse attack. When all nodes in the network perform this auditing routinely, attackers are discovered and can be removed from the neighbor sets of correct nodes. Several defense strategies are discussed in the literature that require additional constraints on neighbor selection [23, 71].

In **file poisoning**, attackers [26, 94, 95] try to inject useless data (poison) into the system. The goal of this attack is to replace a file in the network by a false one. However, when a polluted file is downloaded by an user, it stays available for a while before being inspected and cleansed. After a period of time, all polluted files are eventually removed and the authentic files become more abundant than the corrupted ones.

Cornelli et al. [33] introduced another class of security threat where p2p applications can be exploited to distribute malicious software, such as viruses and trojan horses. In fact, shared audio and video files may harbor security threats, as the multimedia formats permit the introduction of links and active content that may be exploited to introduce malicious software into a computer [13]. In order to combat the threat against suspicious shared software and resources, Damiani et al. [37] proposed defence strategies that uses combined reputations of servents and resources, providing more informative security strategy. Resource reputations are tightly coupled to the resources' content via their digest, thus preventing their forging on the part of malicious peers.

Another important issue that looms over p2p networks is blocking and throttling of p2p traffic by the Internet service providers. According to a 2007 Internet study, 69% of Internet traffic in Germany is p2p traffic, with HTTP way behind at 10% [148]. Given the staggering proportion of Internet traffic accounted for by p2p applications, it is not surprising that ISPs are starting to block ports on which well-known file sharing applications run. For example, Comcast recently started to throttle and drop packets of BitTorrent traffic, effectively blocking its customers from running the software [153]. Going even further, Ohio University recently started to block all p2p traffic on its campus [154].

2.2.3 Network stability in the perspective of complex networks

Since large scale p2p networks can be modeled as complex graphs, stability analysis of complex theoretical graphs, simultaneously done by the physicists and mathematicians provide rich input towards understanding the various properties of p2p networks. In order to analyze the network stability, we need to define proper stability metric. Hence in this section, we first provide extensive review on stability metric. A review on the effect of failures and attacks on network stability is elaborated next. Finally we report the resilience and other related properties of some typical topologies and real world networks.

Stability metrics

There are various approaches to measure the stability of large scale networks. We may characterize these stability metrics in two different categories; metrics based on the (a) change in the topological properties (b) identification of network breakdown point.

(a) Metrics based on the change in topological properties: To measure the networks' error tolerance, Albert et al. [9] studied the changes in the diameter, largest component size, and average component size when a fraction f of the nodes are removed. The absence of a node in general increases the distance between the remaining nodes, as well as the diameter, since it can eliminate some paths that contribute to the system's interconnectedness [97]. In addition, when nodes are removed from the network, clusters of nodes, whose links to the system disappear, can get detached from the main component. The size of the largest connected component S, also termed as giant component, can be used as the stability metric, when a fraction f of the nodes are removed [28]. It has been observed that as the fraction of nodes removed f increase, S displays a threshold-like behavior such that for $f > f_c$, S becomes 0. A similar metric is to monitor the average component size $\langle s \rangle$ of the isolated clusters (i.e. all the clusters except the largest one). For small f, only single nodes break apart, hence $\langle s \rangle \simeq 1$. But as f increases, the size of the fragments that fall off the main cluster increases. At f_c the system practically falls apart, the main cluster breaking into small pieces, leading to $S \simeq 0$, and the size of the fragments, $\langle s \rangle$ peaks. As we continue to remove nodes further ($f > f_c$), these isolated clusters fragment, leading to a decreasing $\langle s \rangle$.

In [166], Wanga et al. introduced a new metric namely network efficiency E, which is based upon the inverse of the shortest distance between nodes, such as

$$E = \frac{1}{N(N-1)} \sum_{i \neq j} \frac{1}{d_{ij}} \tag{2.4}$$

High efficiency network means that pairs of nodes are on average close to each other. This is very similar to the average distance but allows to consider disconnected networks. In a similar fashion, [132] introduced the Diameter-Inverse-K (DIK) measure defined as $\frac{d}{K}$, where d is the average distance between pairs of connected nodes, and K is the fraction of pairs of nodes which are connected. This measure can identify the amount of disconnectedness as well as can differentiate between connected graphs having short or large average node distances.

Recently, a new measure of fragmentation has been developed in social network studies [25]. Suppose a fully connected network of N nodes is fragmented into m separate clusters by removal of nodes. The degree of fragmentation F of the network is defined as the ratio between the number of pairs of nodes that are not connected in the fragmented network to the possible number of pairs in the original fully connected network. Suppose there are m clusters in the fragmented network; since all members of a cluster are, by definition, mutually reachable, the measure F can be written as follows

$$F = 1 - \frac{\sum_{j=1}^{m} N_j(N_j - 1)}{(N(N-1))} = 1 - C \tag{2.5}$$

Here, N_j is the number of nodes in cluster j, and N the number of nodes in the original fully connected network. For an undamaged network, $F = 0$. For a totally fragmented network, $F = 1$. The quantity C can be regarded as the "connectivity" of the network. In this paper, Chen et al. studied the statistical behavior of $F(\equiv C)$ using both analytical and numerical methods and relate it to the traditional measure, the relative size of the giant component S, used in percolation theory.

(b.) Metrics based on the identification of network breakdown point : The stability of networks is also measured in terms of a certain fraction of nodes called *percolation threshold* [14, 22, 101, 134] removal of which breaks down the network into large number of small, disconnected components. Below that threshold, there exists a giant component which spans the entire network. The critical condition for the formation of the giant component in random graphs is described in [115, 116]. These papers of Molloy et al. have theoretically shown that the existence of giant component can be mathematically captured by the ratio $\kappa = \langle k^2 \rangle / \langle k \rangle$ where $\langle k \rangle$ and $\langle k^2 \rangle$ are the first and second moments of the degree distribution respectively; the value of $\kappa \geq 2$ indicates the situation where stability of the giant component is maintained. It is important to note that, for a given finite size network, the notion of percolation threshold does not make sense: the fraction of nodes in the largest connected component will never be zero. Hence, we broadly categorize the procedures to calculate percolation threshold in two different ways

(i) Identifying percolation threshold by observing the topological properties at breakdown point : One may notice that, when the network reaches the threshold point (of breakdown), the slope of the largest connected component size as a function of the number of removed nodes goes to zero. In finite-size computation, we may therefore consider that we reach the percolation threshold when this slope is maximal [22]. In another approach [70, 101], the percolation threshold is detected when the largest connected component size is less than 5% of the whole network.

(ii) Identifying percolation threshold based on classical theories:

In [134], Paul et al. obtained the percolation threshold from simulation following the classical theory of giant component. The nodes were deleted in the network (of size N) following a specific strategy and after each node removal, they calculated $\kappa = \langle k^2 \rangle / \langle k \rangle$. When κ becomes just less than 2 they recorded the number of nodes f, removed up to that point. This process is performed for many realizations. The percolation threshold f_c is calculated as

$$f_c = \frac{\langle f \rangle}{N} \tag{2.6}$$

In [55], a similar procedure is used to calculate percolation threshold. Here a fraction of nodes f are successively removed along with the adjacent links. After each removal, moments of the degree distribution $\langle k^2 \rangle$ and $\langle k \rangle$ are calculated. If $\kappa = \langle k^2 \rangle / \langle k \rangle > 2$, a giant component spans over the network. This procedure is repeated for a large number of realizations (typically 100-300). For each fraction f of removed nodes, the probability F_∞, that a spanning cluster does not exist is calculated. The percolation threshold f_c is that value of f at which F_∞ crosses 0.5.

In summary, we have illustrated several techniques available in the current literature to measure the network stability. However, we find that computing percolation threshold based on the classical theories [55] is the most recent technique and it nicely captures the theoretical foundation of percolation threshold (κ) in the simulation environment. Hence in this thesis, we use this versatile technique to measure percolation threshold during simulation based experiments.

Resilience against failure and attacks:

Albert et al. [9] experimentally addressed the question of random failures and intentional attacks on wide variety of networks. In the case of random failure, nodes are removed randomly irrespective the degree; however in intentional attack, a fraction of high degree nodes are removed from the network. Simulation results in [9] showed that scale free networks display a high degree of resilience against random failures, a property not shared by E-R graph. They argued that this is the basis of the error tolerance of many complex systems like Internet and other communication networks; while key components regularly malfunction, local failures rarely lead to the loss of the global information-carrying ability of the network. Their simulation results also suggested that unlike random failure, scale free networks are highly sensitive against intentional attacks. The diameter of these networks increases rapidly and network breaks into many isolated components when the most connected nodes are attacked. But for E-R graph, there is no substantial difference whether the nodes are selected randomly or decreasing order of connectivity. These two opposite behaviors of E-R and scale free networks are due to their homogeneous and inhomogeneous connectivity distribution respectively. In [28], Cohen et al. analytically calculated the percolation threshold for generalized random graphs against random breakdown of nodes. They considered E-R network and scale free networks as special cases and calculated the percolation threshold for these two graphs. According to their analysis, giant component of E-R graph dissolves when average degree $\langle k \rangle$ is less than or equal to one. In the case of Internet which follows power law distribution, percolation threshold depends on the exponent α. For $\alpha > 3$, there exits a percolation threshold f_c which fragments the giant component. However it has been shown that for $\alpha < 3$, $f_c \to 1$, which indicates that giant component exists for the breakdown of arbitrary large fraction of nodes. In finite systems, transition is always observed, although for $\alpha < 3$, the percolation threshold becomes exceedingly high. For example, considering the enormous size of Internet ($N > 10^6$) one needs to destroy 99% of the nodes before the giant component actually gets destroyed. Subsequently, in [29] Cohen et al. studied the problem of intentional attack in scale free networks. According to their analysis, since just a few nodes of very high degree control the connectivity of the entire system, very few fraction of nodes is needed to be removed to destroy the giant component. They analytically showed that percolation phase transition exists for all scale free networks having $\alpha > 2$. Callaway et al. [22] introduced the concept of percolation process and applied it to examine the resilience of various real world networks like Internet. Using the generating function formalism, they found the exact analytic solutions for node percolation in scale free networks. In addition to uniform failure, they also formally modeled the intentional attack by Heaviside step function. They showed both analytically and experimentally the change in the giant component size against attack as the function of cut-off degree and fraction of node removed. In [127], Newman et. al have developed the theory of random graphs with arbitrary degree distribution with the help of generating function formalism. This paper analytically derived the exact condition of phase transition towards formation of giant component, expressions to calculate average component size, the size of the giant component if there is one and the average node-to-node distance within the graph. The results were compared with several real world networks like WWW, collaboration network etc to demonstrate the accuracy of their analysis.

There is a huge amount of work done in [54, 55, 57] to study the robustness of scale free networks under systematic variation of attack strategies. Gallos et al. [54, 55, 57] introduced a general attack strategy where the probability that a given node is removed, depends on the number of its links k via

$$W(k_i) = \frac{k_i^\alpha}{\sum_{k=1}^{N} k_i^\alpha} \tag{2.7}$$

For $\alpha > 0$, nodes with larger k are more vulnerable, while for $\alpha < 0$, nodes with lower k are more vulnerable. The limiting cases $\alpha = 0$ and $\alpha \to \infty$ represent the random removal and targeted attack respectively. The results showed that the critical fraction f_c needed to disintegrate the network increases monotonically as α decreases. The work is extended in [55], where the attack strategy is associated with the knowledge available to the attackers. For example, in the intentional attack, removal of only a small fraction of nodes is sufficient to destroy the network. This strategy, however, requires full knowledge of the network topology in order to identify the highest connected nodes. In many realistic cases, this entire information is not available, and only partial knowledge exists. Accordingly, the high degree nodes can be removed only with a certain probability that will depend on k. [55] showed that even a little knowledge of the highly connected nodes in an intentional attack reduces the threshold drastically compared with the random failure. This pointed to the vulnerability of the Internet which can be damaged efficiently when only a small fraction of hubs is known to the attacker. Moreover, this result is also relevant for immunization of populations; even if the virus spreaders are known with small probability, the spreading threshold can be reduced significantly. They also showed that even if the attack does not yet disintegrate the network, there is nevertheless a major damage on the network, since the distances between the nodes increase significantly and any transport process on the network may become inefficient. Holme et al. [73] studied the response of complex networks subject to attacks on nodes and edges. Several existing complex network models as well as real-world networks of scientific collaborations and Internet traffic were numerically investigated, and the network performance was quantitatively measured by the average inverse geodesic length and the size of the largest connected subgraph. For each case of attacks on nodes and edges, four different attacking strategies were used: (A) removal by the descending order of the degree, (B) betweenness centrality and these parameters are calculated either from the initial network (A_1, B_1) or from the current network (A_2, B_2) during the removal procedure. The results identified that removals by the recalculated degrees and betweenness centralities are often more harmful than the attack strategies based on the initial network suggesting that changes in the network structure is important as nodes or edges are removed.

Lathapy et al. [70] investigated the often claimed affirmation that successful attacks can be launched on scale-free networks, because large number of links get removed as soon as the top degree nodes are removed. They showed that removing the same number of links at random has much less impact, showing that this (removal of links) was not the sole reason behind the attack efficiency. Finally, they proposed two new node/link removal strategies and compared them with classical attack where high degree nodes are removed. The first failure strategy randomly removes the nodes of degree greater than one. This decreases the number of nodes of degree higher than 1 and increases the number of nodes of degree 0 or 1. The results revealed that this kind of failure gives almost similar impact on networks compared to the classical attacks. This tends to show that the presence of a threshold for classical attacks is not due to a high efficiency, but rather to the fact that they do not remove nodes of degree 1. The second failure strategy they proposed was based on link removal; they removed, at random, links between nodes with degree greater than 1. They claimed that this failure strategy is more efficient than the classical attack, with respect to the fraction of removed links. In classical attack strategy, one may remove many links attached to nodes of degree 1, which does not help in destroying the network. This strategy, on the opposite, focuses on those links which really disconnect the network. Crucitti et al. [36] studied the effects of errors and attacks on the efficiency of scale-free networks. Two different kinds of scale-free networks have been considered and compared to random graphs: scale-free networks with no local clustering produced by the Barabasi-Albert (BA) model, and scale free networks with high clustering properties as in the model by Klemm and Eguíluz (KE) [86]. They investigated the effects of errors and attacks both on the global and on the local efficiency of the network. The measure of network efficiency is defined by Wanga [166] in Eq. (2.4). The global efficiency signifies the connectivity of the overall network whereas local efficiency denotes the connectivity among the subcomponents of the network. The results showed that both the global and the local efficiency of scale-free networks are unaffected by the failure of some of the nodes. On the other hand, in scale-free networks the global and the local efficiency rapidly decrease when the nodes removed are those with higher connectivity. These properties are true both for BA networks and for KE networks, though KE networks have higher local efficiency but lower global efficiency than BA networks.

Resilience of some typical topologies

Tanizawa et al. [161, 162] provided a set of network design guidelines which maximized the robustness of the scale free networks both to random failures of nodes and attacks targeted on the highest degree nodes. They examined the stability of two regime power law networks, networks with combined degree distribution of power law and exponential, two Gaussian distribution etc. Percolation on random regular graph is discussed in [10, 120]. The works done in [119, 172] showed that in many physical networks, the removal of selected nodes can have a much more devastating consequence when the intrinsic dynamics of flows of physical quantities in the network is taken into account. In a power transmission grid, for instance, each node (power station) deals with a load of power. The removal of nodes, either by random breakdown or intentional attacks, changes the balance of flows and leads to a global redistribution of loads over all the network. Subsequently, Crucitti et al. [35] introduced a simple model to explain the possibility of rare but catastrophic effect, triggered by small initial shocks, present in most of the complex communication/transportation networks. The results showed that the breakdown of a single node is sufficient to affect the efficiency of a network up to the collapse of the entire system if the node is among the ones with largest load. This is particularly important for networks with a high hetereogeneous distribution of node loads like scale-free networks as well as real-world networks like Internet and electrical power grids.

Specialized network: Along with generalized networks, resilience of some of the more uncommon and specialized topologies has been discussed in recent days. In [133, 135, 161, 162], Paul et al. introduced the concept of bimodal networks. They proposed the guidelines to maximize the robustness of various kinds of networks (single regime power law, two regime power law, bimodal networks) to both random failure and intentional attack. In the similar line, Valente et al. [164] showed analytically that the network configurations that maximize the percolation threshold under attack and/or random failures have at most three distinct node degrees. Recently, Paul et al [134] used Monte Carlo simulations to calculate percolation threshold in bimodal networks and shown that the general criterion for percolation stated in [28] becomes invalid for most of the real world networks due to the presence of degree-degree correlation.

Degree correlated network: A study on the real-world networks revealed that most of these networks are degree correlated [123, 125]. Social networks are found to be assortative (higher high-high degree connections) whereas networks like information networks, technological networks, biological networks are found to be disassortative (very less high-high degree connections). Noh [128] investigated the nature of the percolation transition in a correlated network with a Poisson degree distribution. First, he proposed a model for network generation with a tunable degree correlation for a given degree distribution. The results revealed that negative correlation is irrelevant since the percolation transition in the disassortative network belongs to the same universality class as in the uncorrelated network. However, the impact of the positive correlation on the percolation transition is relevant. In the positively correlated network, even at the critical point, mean size of finite components does not diverge, hence a non-percolating phase transition occurs in the network. In [165], Vazquez et al. studied the effect of degree correlations considering some examples of uncorrelated, assortative, and disassortative graphs. They derived some general expressions to obtain the bounds on the percolation threshold in correlated networks. Their results showed that the existence of a finite amount of random mixing of the connections between nodes is sufficient to make the graph robust under node or edge removal provided the second moment diverges. Assortative correlations make the situation even better; they can make a graph robust to random damage, even with a finite second moment of the degree distribution. This is in contrast to uncorrelated networks, which are robust only if the second moment of a degree distribution diverges [28]. Recently, Goltsev et al. [67] also studied the percolation transition in complex networks with degree-degree correlations. They demonstrated that both assortative and disassortative mixing affect not only the percolation threshold but can also change critical behavior at this percolation point. Their analysis showed that the critical behavior is determined by the eigenvalues of 'branching matrix' and a degree distribution. They derived the necessary and sufficient conditions for a correlated network to have the exactly same critical behavior as an uncorrelated network with the identical degree distribution. According to their analysis, a network may be robust against a random failure even if the second moment of its degree distribution is finite. On the other side, specific disassortative network may be fragile even when the second moment of the degree distribution is divergent.

Real world networks: Huang et al. [74, 75] presented a detailed and in-depth study on the response of peer-to-peer networks subject to attacks, and investigated how to improve attack survivability by properly modifying their topological properties. They revealed the topological weaknesses of Gnutella-like p2p networks [80] by identifying attack vulnerability via extensive simulations under realistic operating conditions. The results indicated that these networks are extremely robust to random failures whereas highly vulnerable under intentional targeted attacks, which is consistent with classical theories. Ripeanu et al. [142] showed that Gnutella node connectivity follows a multi-modal distribution, combining a power law and a quasi-constant distribution. This property keeps the network as reliable as a pure power-law network when assuming random node failures, and makes it harder to attack by a malicious adversary. They also suggested few precautions for Gnutella network to ward off potential attacks. For example, the network topology information, that can be obtained easily permits highly efficient denial-of-service attacks. Hence, some form of security mechanisms that would prevent an intruder from gathering topology information appears essential for the long-term survival of the network (although it would make global network monitoring more difficult if not impossible). They also designed an agent that constantly monitors the network and intervenes by asking servents to drop or add links as necessary to keep the network topology optimal.

Network immunization: Percolation in random graph is also used to model the spread of diseases or computer viruses where some of the nodes are occupied by the disease/virus and others are not [124]. The spreading dynamics are closely related to the structure of networks. Emergence of giant component may spread the diseases/viruses in the network which triggers epidemics in the society or Internet. Various immunization techniques have been proposed to arrest the spread of disease or computer viruses in the network [30]. Homle et al. [72] investigated strategies for vaccination and network attack that are based only on the knowledge of the neighborhood information. The analysis revealed that for most networks, regardless of the number of vaccinated vertices, the most efficient strategies are to choose a vertex v and vaccinate a neighbor of v with highest degree, or the neighbor of v with most links out of v's neighborhood. In this vaccination process, the node v can be either the most recently vaccinated vertex (chained selection) or any random vertex (unchained selection). For real-world networks the chained versions tend to outperform the unchained ones, however the unchained strategies are preferable for networks with a very high clustering or strong positive assortative mixing (larger values than in seen in real world networks). In summary, choosing the people to vaccinate in the right way will save a tremendous amount of vaccine and side effect cases. Huang et al. [75] developed a novel framework for better characterizing the immunization of Gnutella-like p2p networks by taking into account the cost of curing infected peers. They prefer a small fraction of the most high-degree nodes as targets to inject immunization information and the immunization processes probe the network in a parallel fashion along links that points to a high-degree node with a probability proportional to k^{α}.

2.2.4 Scopes of work

A detail review of complex network theoretic approaches reveals that a wide variety of work has been done on the stability analysis of large scale dynamic networks. In [28] and [29], Cohen et al. calculated the deformed degree distribution of the scale free network after random failure and intentional attack. However, a more complete theory is needed to calculate the degree distribution of a network after any arbitrary kind of attack. This may enable us to achieve a more generalized framework to predict the exact point of breakdown of a network after an arbitrary attack. Most of the theories in the literature deals with the percolation in an infinite size networks. However, the impact of network size on the percolation threshold need to be investigated. In p2p literature, various churn models have been proposed and some measurement based results have been reported from rigorous experiments. Several network attack strategies and defence techniques are also discussed. However, understanding the exact impact of churn and attacks on the superpeer topology as well as analyzing the influence of various structural parameters (like peer-superpeer fraction, their individual fraction, degree correlation present in the network) on network stability is essential, but not systematically investigated till now. A thorough theoretical understanding in this aspect may in turn enable network engineers to properly interpret the measurement results and design optimum network topology to improve various p2p services.

2.3 Dynamics of peer-to-peer networks

P2p networks are formed and maintained amidst continuous node and link dynamics like bootstrapping of new nodes, peer churn, link rewiring etc. In subsection 2.3.1, we review several proposed and commercially used bootstrapping protocols. This is followed by a survey on the complex network theoretic approaches on network growth. The impact of several link dynamics in p2p networks is reported in section 2.3.3.

2.3.1 Bootstrapping protocols

The key operation in any peer-to-peer network is bootstrapping, the initial discovery of other online nodes participating in the network. Nascent peers need to perform such an operation in order to find the IP addresses of online peers and connect to them. The joining peers execute a bootstrapping function through peer servents like Limewire, Gnucleus etc (in case of Gnutella network). The bootstrapping protocols play a major role in the construction of efficient network topology and have significant impact on the performance of the p2p networks. Significant amount of technical works [31, 62, 78, 82, 130, 156, 169] have proposed construction of optimum peer-to-peer topology aimed towards maintaining desired quality of services. The proposed bootstrapping protocols mainly aim to satisfy individual optimization criterions that often land to the conflict of interests. The initial neighbors of a joining peer (selected by bootstrapping) play an important role on the QoS enjoyed by that peer. These initial neighboring peers determine the new peer's location in the network topology, and eventually its search and download efficiency. For instance, connecting with a well networked peer possessing high processing power, large storage space put the new node at the center of the topology thus reducing search latency and file download time [82]. On the other hand, the time spent by the peer in bootstrapping is critical because until the bootstrapping step is completed, a peer cannot participate in the file sharing activities. Hence, a school of research proposed simplified joining protocols to minimize the bootstrap latency. Recent research focuses on improving the performance of p2p networks by incorporating network and semantic awareness in the bootstrapping process [34]. In these protocols, the nodes with close physical proximity and similar interests are selected for the initial connection establishment. Network diameter is also considered as an important optimization metric; some bootstrapping protocols are proposed which result in a provably strong guarantees on the maximum network diameter [130]. Cramer et al. [34] compared different bootstrapping techniques for p2p networks, including static bootstrap servers, out-of-band node caches, random address probing, and network layer mechanisms etc. A brief overview of these bootstrapping protocols follows next.

Random Address Probing

This is possibly the most simple bootstrapping technique suitable for large-scale overlay networks. A node wishing to join the network, randomly draws an IP address from the global (or local) address space. It then tries to establish a connection to this IP address using a well-known port. In case the connection cannot be established because the node does not exist or does not participate in the p2p overlay, another address has to be tried. Experimental results showed that a brute-force random global scan for Gnutella peers requires on average 2425 attempts before finding the first peer. While random address probing may be successfully used by very large p2p networks, it is not efficient for small to medium scale networks, as many probes are necessary. In [62] GauthierDickey et al. proposed a fully distributed bootstrapping of peer-to-peer networks which generates a stream of promising IP addresses to be probed as entry points. In [31], Conrad et al. proposed a generic, distributed and self-organizing bootstrap service based on the random address probing. The proposed bootstrap service itself uses a p2p network namely 'bootstrap p2p network', for distributed storage of the bootstrapping information. The fundamental assumption is that, IP addresses in peer-to-peer networks have a significant bias in their distribution across different organizations, as evidenced in Gnutella and Skype measurements. Hence, the protocol is based on the classification of IP address ranges across the organizations using DNS that may help to improve the success rates of bootstrapping. The paper claimed that this approach improved the 'success rate' of random address probe for small private p2p networks as well as for large p2p networks.

Employing Network Aware Mechanisms

In order to optimize the traffic in the physical network, the discovery of online nodes while bootstrapping, should be based on the topological structure of the underlying network. If some online nodes are available in the same network segment, it is both convenient and highly efficient to use network layer mechanisms to connect to these nodes, at least for bootstrapping. In this context, Cramer et al. [34] proposed locality aware bootstrapping service that supports p2p overlay networks in discovering nodes that are nearby in terms of the underlay network topology.

Modifications for improving QoS

One of the most cited papers in the topic, Pandurangan et al. [130] proposed a bootstrapping algorithm which builds a p2p network with small diameter keeping the average degree constant. However, the design heavily depends on a central server that is needed to coordinate the connections among peers. In [168, 169], Wouhaybi et al. proposed Phenix, a distributed algorithm that constructs scale free p2p networks offering resilience and fast response time to users. The algorithm used some variations of preferential attachment along with some amount of randomness mixed within it. The work done in [78,117] proposed gossip based protocols to construct the application specific superpeer networks. The idea of a generic bootstrap service in structured p2p networks was also discussed in [24] and [77]. Recently, in [90] Kwong et al. proposed a distributed bootstrapping protocol with random walk based joining and relinking process and have shown that topological structure of p2p networks depends heavily on the node heterogeneity and capacity distribution of joining nodes.

Bootstrapping Servers

Initially, in p2p networks like Gnutella, the bootstrapping problem was solved by placing static nodes (i.e. servers) in the overlay. Peer servents contained hard-coded DNS names of the bootstrapping nodes. These bootstrapping nodes, called 'pong caches' [81] in Gnutella, collected topological information and addresses of other nodes participating in the overlay. On request by a newly joining node, the pong caches returned a list of addresses of nodes seen recently in the overlay. The new node then tried to establish overlay connections to the nodes in this list. This bootstrapping method though simple however lacked scalability [167] and required at least some administrative overhead.

Out-of-band node caches: GWebCache

The Gnutella community introduced GWebCache (Gnutella web caches), to overcome the limitations of the pong cache mechanism [82]. The GWebCache system functions as a distributed repository of the online peers in the network. There are two fundamental activities that is associated with the GWebCache (i) accessing the cache (ii) updating the cache.

(i) Accessing the cache: When a new peer wants to join the Gnutella network, it can retrieve the host list from one or more of these GWebCaches. Limewire and Gnucleus maintain a separate list of superpeers and give priority to hosts in this list during connection initiation. Since superpeers have relatively long uptimes and the capability to support more incoming connections, prioritizing these peers during connection initiation increases the probability of successful connections hence reduces the bootstrap latency. The GWebCaches also maintain a list of other web caches in the network.

(ii) Updating the cache: A host accepting incoming connections, updates the GWebCache with its own IP address and port number, and with information about some other GWebCache that it believes is alive. Some of the peer servents like LimeWire and Mutella update the GWebCaches only in the superpeer mode. The peers in the Gnutella network are responsible for keeping the information in these caches up-to-date; the caches do not communicate with each other at any time.

It becomes quite evident that GWebCache based bootstrapping protocols aim to select resourceful nodes for the initial connection establishment. This optimizes the search latency and file download time of the newly joined peer due to the efficiency of its neighbors. GWebcache based bootstrapping protocols are widely used by the servents of the popular superpeer networks, like Gnutella 0.6. Recent study [2] revealed that there are 20 GWebCaches run by various servents, taking more than $700,00$ requests per hour, which is almost 200 requests per second. Hence in this thesis, we concentrate on GWebCache based bootstrapping protocols and investigate the formation of superpeer nodes in the network. We find that the GWebcache based bootstrapping protocols can be suitably modeled by the preferential node attachments rules where preference is given on the good nodes possessing high bandwidth and processing power. There is a great deal of interest among physicists and mathematicians in understanding the growth of networks following preferential attachment [11,16,18,87]. A detailed review helps us to understand the problem in context and to develop suitable modeling scheme for explaining the emergence of superpeer network.

2.3.2 Network growth in the perspective of complex network theory

It has been found that degree distribution of most of the large scale networks like Internet, Web network etc follow power law, hence most of the theories developed to understand the topology of the emerging network is directed towards explaining the emergence of scale free networks. In this context, several complex network theoretic models are proposed in the literature. The most popular among them is the Barabasi and Albert **(BA) model** where appearance of scale free networks is explained with the help of preferential attachment rules [11]. Kleinberg et al. proposed **vertex coping** model of network growth [84, 89] where the network grows stochastically by constant addition of nodes and the addition of directed edges either randomly or by replicating the edges from another existing node. In [51], Fabrikant et al. proposed a plausible explanation of the power law distributions observed in the graphs arising in the Internet topology. In this growth paradigm (**FKP growth model**), the incoming node i stochastically connects to an existing node j such that the node j is physically close to node i (small Euclidean distance d_{ij} between the node i and j) and at the same time the node j is centrally located in the network (hop distance of j, (h_j) to the other nodes is minimum). Callaway et al. [21] proposed the model of evolving network that is initially scattered into disconnected components and eventually merged with each other to form large components. In this model, nodes may join the network without necessarily connecting with some existing node. Then, with probability d, two nodes are chosen uniformly at random and joined by an undirected edge. This may result in a growing network containing isolated nodes along with components of various sizes. However, among these various growth models, the BA model is the simplest one, widely studied and suitable to model the GWebCache based bootstrapping process. Hence, in this section we provide a detailed review on BA model and its wide range of variations.

Barabasi–Albert model and its variations

The original concept of scale free network and preferential attachment was discovered by Derek de Solla Price. In 1965, he described the first instance of a scale-free network in the citation network. He found out that both the in-degrees and out-degrees of the network of citations between scientific papers follow a power-law distribution [39]. Some years later, Price published his explanation for the arising power-law degree distributions [40]. He built up his work on the ideas of Herbert Simon [150], developed in 1950s, which showed that power law arises when "the rich get richer". Price coined the term *cumulative advantage* to denote this phenomenon. However the work was rediscovered some decades later by Barabasi and Albert [11], who gave the attachment a new name - *preferential attachment*. In this highly influential paper, Barabasi et al. proposed a network growth model of the World Wide Web network. They showed that, a power-law degree distribution emerges naturally from a stochastic growth process in which new nodes link to existing ones with a probability proportional to the degree of the target node. The model of Barabasi and Albert (also popularly known as BA model) attracted an exceptional amount of attention in the literature. In addition to analytic and numerical studies of the model itself, many researchers have suggested extensions or modifications of the model that alter its behavior or make it a more realistic representation of processes taking place in real-world networks [87]. The refined variants of this preferential attachment process allow nonlinear attachment probabilities, fitness of nodes and links, aging, rewiring of edges, and appearance of truncated power law [51, 87, 155]. A brief review follows.

Nonlinear attachment: In [87], Krapivsky et al. generalized the BA model where the probability of attachment to a node goes as some general power of degree k^{γ}. By solving the model by rate equations, the paper found three general classes of behavior. In linear preferential attachment ($\gamma = 1$), the evolving network showed power-law degree distribution. In sublinear attachment ($\gamma < 1$), the degree distribution became power law multiplied by a stretched exponential, whose exponent is a complicated function of γ. In superlinear case ($\gamma > 1$) a 'condensation' phenomenon was identified, in which a single node gets a finite fraction of all the connections in the network, and for $\gamma > 2$ there is a non-zero probability that this "gel node" gets connected to every other node on the graph. In the same article, Krapivsky et al. showed that there is a correlation between the age and degree of the nodes; older nodes acquired higher mean degree. For $m = 1$, the degree distribution of a node i with age a may be expressed as

$$p_k(a) = \sqrt{1 - \frac{a}{n}} \left(1 - \sqrt{1 - \frac{a}{n}}\right)^k \tag{2.8}$$

Thus for specified age a, the distribution is exponential, with a characteristic degree scale that diverges as $\sqrt{1 - \frac{a}{n}}$ as $a \to n$. The older vertices have substantially higher expected degree than the vertices added later, and the overall power-law degree distribution of the whole graph is a result primarily of the influence of these earliest vertices. In [42, 43], Dorogovtsev et al. also took the ageing of nodes into account so that a link is connected not only preferentially towards degree, but also depending on the age of the node. In [43], each new node of the network is connected to some old node with probability proportional (a) to the connectivity of the old node as in the Barabasi-Alberts model and (b) to $\tau^{-\gamma}$, where γ is the age of the old node. The simulations and theoretical results revealed that the network shows scale free behavior only in the region $\gamma < 1$. When γ increases from $-\infty$ to 0, the power law exponent of the degree distribution grows starting from the value 2. At $\gamma = 1$, the exponent moves to ∞ whereas for $\gamma > 1$, the distribution p_k becomes exponential. The correlation between the degree scaling and age of the vertices has been used by Adamic and Huberman [5] to show that in actual World Wide Web network data, there is no such correlation present as such. It seems that the dynamics of the Web is much more complicated than this simple model can explain. Adamic and Huberman suggested that, this is because the degree of vertices is also a function of their intrinsic worth; some Web sites are useful to more people than others and so gain links at a higher rate. This gives rise of the concept of 'fitness' that represents the attractiveness of a node to accrue new links.

Fitness model: Bianconi and Barabasi [18] argued that in real networks, nodes have a competitive aspect such that each node has an intrinsic ability to compete for edges at the expense of other nodes. This paper proposed a model in which each node is assigned a fitness parameter η which does not change over time. Thus at every timestep a new node j with a fitness n_j is added to the system, where j is chosen from a distribution $\rho(\eta)$. Each new node connects to m existing nodes in the network, and the probability to connect to a node i is proportional to the degree and the fitness of node i. The rate of change of degree of a node can be calculated with the help of continuum theory [11]. The results offered interesting insights into the evolution of nodes in a competitive environment. In the scale-free model, where each node has the same fitness, all nodes increase their connectivity following the same scaling exponent $\beta = 1/2$. In contrast, when different fitness is allowed, multiscaling emerges and the dynamic exponent depends on the fitness parameter, η. This allows nodes with a higher fitness to enter the system at a later time and overcome nodes that have been in the system for a much longer timeframe. It is interesting to note that despite the significant differences in their fitness, all nodes continue to increase their connectivity following a power-law in time. Thus, the results indicated that the fitter wins by following a power-law time dependence with a higher exponent than its less fit peers. A number of variations on the fitness theme of Bianconi-Barabasi model have been studied by Ergun and Rodgers [50]. This paper proposed a model where instead of multiplying the attachment probability, the fitness η contributes additively to the probability of attaching a new edge to node i. Treating the models analytically, Ergun et al. found that for suitable parameter values, the power-law degree distribution is preserved, although the exponent may be affected by the distribution of fitness, and in some cases there are also logarithmic corrections to the degree distribution. The impact of the fitness distribution $\rho(\eta)$ is also analyzed in [17,88]. The results indicate that depending on the distribution $\rho(\eta)$, the network either shows a power-law degree distribution or one node with the highest fitness accrues a finite fraction of all the edges in the network; a sort of "winner takes all" phenomenon.

In [45], Dorogovtsev and Mendes extended the BA model in which the probability of attachment to a vertex of degree k is proportional to $k + k_0$ where offset k_0 represents the amount of randomness incorporated within the attachment kernel. Extensive analysis of the model revealed that this kind of attachment generates power law degree distribution ($p_k \sim k^{-\alpha}$) for large k, with exponent $\alpha = 3 + k_0/m$. Further analysis revealed that negative values of k_0 could be the explanation of the power law exponent $\alpha < 3$ seen in real-world networks. In [44], Dorogovtsev et al. proposed more sophisticated BA model whereby edges appear and disappear between preexisting nodes with stochastically constant but possibly different rates. They found that over a wide range of values of the rates, the power-law degree distribution is maintained, and the exponent varies in the zone $\alpha < 3$ based upon the rate. In another article, Bianconi [16] presented a growth framework in which both nodes and links are assigned some weights. Two classes of weighted networks are considered (i) class I, in which the degree of the node does not affect the weights of its links, (ii) class II, in which the node degrees strongly influence the link weights. In this framework both node degrees and link weights increase following a preferential attachment rule. The strength of a node i in the weighted network is defined as the sum of all the weights of incoming and outgoing links $s_i = \sum_j w_{ij}$. The results showed that networks of class II emerge only when the rate, at which nodes are strengthened is higher than the rate at which new links are established.

Krapivsky and Redner [88] studied a full directed-graph model in which both vertices and directed edges are added at stochastically constant rates and the outgoing end of each edge is attached to vertices in proportion to their out-degree and the in-going end in proportion to in-degree, plus appropriate constant offsets. [88] showed that this kind of dynamics give rise to power law in both the in and out-degree distributions, just as observed in the real Web network. By varying the offset parameters for the in and out-degree attachment mechanisms, one can even tune the exponents of the two distributions to agree with those observed in the Web network. Some of the works related to the presence of exponential cutoff degrees in the WWW networks have been reported in [52,53]. They modeled the appearance of exponential cutoff in the power-law scaling, although this cutoff may only be observable in the tail of the distribution for extremely large networks.

2.3.3 Local events in emerging p2p networks

In addition to the peer bootstrapping, several local events like peer churn and link rewiring play a major role in determining the asymptotic degree distributions and other topological properties of p2p networks. Consequently, efficiency of several functionalities like search also get influenced. We have detailed the survey on peer churn in the previous section. Here we present various strategies of link rewiring taken sometimes proactively, some reactively (to offset churn).

Guclu et al. [68] discovered the relationship between the frequent departure of the online peers and its impact on the search efficiency in p2p networks. This paper developed a rewiring based reactive protocol, activated at the time of node departure due to churn, so that the search performance of the topology remains high. The topology formation and maintenance schemes described in this paper are mainly based upon the local information available. Semantic overlay networks cluster peers that are semantically or socially close into groups, by means of a rewiring procedure that is periodically executed by each peer. This procedure establishes new connections with similar peers and disregards connections to peers that are dissimilar. In [38], Das et al. proposed an algorithm, to identify inherent community edges [126] among peers and evolved the topology accordingly. In that goal, new links were added among similar nodes keeping the original overlay edges intact. The topology evolution algorithm ensured bounded increase in the node capacities (i.e. average number of neighbors that a node can maintain). The simulation results demonstrated that these linking schemes successfully improved the performance of random walk based search algorithms. In [140], a generic approach of rewiring is presented and several variants of this approach are reviewed and evaluated. The results showed the way peer connection is affected by the different design choices across the rewiring mechanisms and consequently the way these choices influenced the overall system performance. In large scale dynamic networks also, a series of microscopic events shape evolution of the network, including the addition or rewiring of edges or removal of nodes or edges. Similar to growth, extensive work has been done to model the churn/rewiring process which we describe next. This is important to note that, in section 2.2.1 we consider churn in a static network, however this section deals with the churn in the continuously evolving network.

Complex network approaches

Several complex network theoretic models have been proposed to investigate the effect of selected processes in various real world networks. Any local change in the network topology can be obtained through a combination of four elementary processes: addition or removal of a node and addition or removal of an edge. But in reality, these events come jointly, such as the rewiring of an edge is a combination of an edge removal and addition. A brief study on all these dynamics follows. First we focus on the addition and removal of nodes and links in a growing and nongrowing network. Then we include the rewiring dynamics in our study.
Krapivsky et al. [12] introduced a simple network growth model with addition and deletion of nodes. In this model, a new node joins the network with a randomly selected existing node. On the other hand, when a node is deleted, its children are attached to its parent. Analysis showed that component size distribution of this kind of evolving network has a power-law tail and the exponent α varies continuously with the addition rate. This paper also observed that the degree distribution evolves by an aggregation process since the parent node inherits all incoming links of a deleted node. Hence, the deletion process results in condensation phenomenon; a giant hub appears which is connected to a finite fraction of the nodes in the system. In [64], Ghoshal et al. focused on creating networks whose topology can be manipulated by adjusting rules of node joining and departure. Moore et al. [118] studied the process of network growth by the constant addition and removal of nodes and edges. The results showed that at steady-state (when node joining and removal rate is almost same), if joining nodes get attached to other nodes at random (without preference), the degree distribution sharply peaked at the maximum and then rapidly decays with Poisson tail. If the joining nodes attach to other nodes using a linear preferential attachment mechanism, the degree distribution becomes a stretched exponential. And finally, when the network shows net growth, i.e. nodes joining faster than its removal, the degree distribution follows a power law with an exponent α such that $3 \leq \gamma < \infty$.

In [7], Barabasi et al. took an important step towards understanding the network evolution incorporating the node and link addition and rewiring of links. Using continuum theory [11, 18] they showed that, depending on the relative frequency of these local processes, networks can develop two fundamentally different topologies. In the first regime, degree distribution p_k has a power-law tail, but the exponent depends predominantly on the relative frequency of the local events. In the second regime the power-law scaling breaks down, and p_k approaches an exponential decay. Finally, they used the model to fit the connectivity distribution of the network describing the professional links between movie actors. Geng et al. [63] introduced the models of network evolution that incorporate the four local processes at every time step: (a) the addition of a new node with new links, (b) addition of new links between old nodes, (c) the rewiring of links and (d) deletion of some existing links. They considered two models (single or double preferential attachment(s)) of evolving networks that give more realistic descriptions of the local processes. The analytical results showed that these two models produce scale free networks ($p_k \sim k^{-\alpha}$) if the parameters are chosen properly. More specifically, the addition of new links between the existing nodes in the network does not change the scaling exponent α. However, the rewiring of links or deletion of existing links decrease α.

In [131], Park et al. showed that a large, non-growing network can evolve by itself into a scale-free state in a self organizing manner. First, in the unweighted model, the investigation starts with a regular network where the links across the nodes were removed and preferentially reestablished constantly in time (which is controlled by the model parameter $0 \leq \lambda < 1$). Analysis showed that a network evolving following such a simple rule can yield a spectrum of degree distributions ranging from algebraic to exponential such that $p_k \sim k^{-\alpha} e^{\xi k}$ where the algebraic exponent α and the exponential rate ξ depend on the model parameter λ. On the other hand, the weighted network model is capable of generating robust scale-free behavior where value of the exponent α in the range that fits many realistic networks (between 2 and 3). Ohkubo et al. [129] applied the model of Park et al. [131] in an undirected network where rewiring probability depends on the fitness parameter.

In [96], Lindquista et al. introduced two different schemes based upon the end of rewiring; (a) rewiring from a randomly selected node and (b) rewiring from a neighbor of the randomly selected node. The equilibrium degree distributions were analytically derived using a general ordinary differential equation (ODE) model. The model provided insights of the net effects of rewiring itself, as the number of links and the number of nodes in the network are conserved throughout the analysis. The results indicated that regardless of the attachment probability (as long as it is same in both rewiring methods), rewiring from a neighbor generally produces more high degree nodes in the equilibrium distribution than rewiring from a random node. Nima et al. [147] used the continuous rate equation approach to predict the power-law exponent in stochastic models, where new nodes preferentially join the network and existing nodes depart the network at a constant rate. Their analysis showed that for such models, the power-law degree distribution appears in the evolving network with exponent $\alpha > 3$ and it rapidly approaches ∞ as the deletion and insertion rates become equal. In the next step, they introduced a compensatory rewiring process, where existing nodes compensate for lost links by initiating new preferential attachments. Further analysis revealed that by regulating the rewiring rate, the exponent of the power-law the degree distributions of the resulting networks (for any deletion rate), can be tuned as close to 0.2.

Networks are formed as a result of many different processes that may have little to do with robustness. It thus seems important to explore whether the existing networks can be modified to improve robustness without appreciably degrading the network's performance. In Beygelzimer et al. [14] proposed such modification schemes (preferential rewiring, preferential random edge rewiring, random edge rewiring, and random neighbor rewiring), wherein either existing edges are randomly rewired to connect different pairs of nodes, or else new edges are added randomly to the network. Such random perturbations decrease the network's dependence on its hubs, making it more robust against degree-based attacks. This paper presented empirical results to show how robustness, as measured either by the size of the largest connected component or by the shortest path length between pairs of nodes, was affected by different strategies that alter the network by rewiring a fraction of the edges or by adding new edges. A modest alteration of an initially scale-free network can usefully improve robustness against attack, particularly when the fraction of attacked nodes is small,

2.3.4 Scope of work

Section 2.3.2 shows that an extensive amount of work has been done in network science to understand the growth of complex networks. However, the main thrust of research still remains in understanding the evolution of scale free networks. Unlike social networks, in computer networks each node can have a maximum connectivity/degree, little attention has been paid to model such constraints. On the other hand, in p2p community, most of the work done are directed towards experimental based understanding of different bootstrapping protocols and to propose improvement in the performance of p2p services. Such ad hoc improvements seem to have limited utility compared to the overhead they incur. Hence, instead of focusing on the more complicated and sophisticated bootstrapping protocols, the impact of simple joining rules on the emerging network topologies need to be understood properly. Analysis of the outcomes of different measurement studies related to the formation of superpeer networks in the light of the complex network theoretic approaches may uncover some interesting results. These results can provide important suggestions to the network engineers to improve the performance of superpeer networks in a very cost effective manner.

2.4 Conclusion

In this survey, we have provided (1) a review on the superpeer networks and the related dynamics that occurs on and of the networks (2) a study of complex network theoretic approaches suitable to capture the topological as well as dynamical aspects of the p2p networks. In category (1), we have presented a substantial review on the topology of peer-to-peer networks, more specifically on superpeer networks. We have discussed the formation of superpeer networks; various proposed and commercially available bootstrapping protocols are presented. The impact of several peer dynamics (user churn and attacks) on the network topology and various defence strategies are also illustrated. In category (2) several models representing large scale complex networks are provided. This is followed by a detailed study on the complex network theoretic approaches related to resilience and evolution of large scale networks.

With a detailed understanding of the state of the art, we move on to report our contributions. In the next two contributory chapters (Chapter 3 and 4), we analyze the stability of superpeer networks against churn and attacks. The Chapters 5 and 6 focuses on the emergence of the superpeer networks amidst bootstrapping and other node and link dynamics.

Chapter 3

Churn and stability of superpeer networks

3.1 Introduction

The following two chapters analyze the stability of superpeer networks against peer churn and attacks. More specifically this chapter deals with the stability analysis against peer churn while Chapter 4 deals with the attacks on the superpeer networks. A detailed background study which brought forward the importance as well as the present state of the art of the problem has been already presented in Chapter 2.

In this chapter, we model superpeer networks with the help of random graphs and peer churn and attacks as the removal of nodes from such network. We represent the network topology as the ensemble of graphs with degree distribution p_k which signifies the fraction of nodes in the network with degree k. Along with modeling the superpeer topology, we also simulate Gnutella networks following (a) commercial servent protocols (b) real data from topological snapshot. The node removal process is based on the node degree and modeled by another probability distribution f_k (probability of removal of a node of degree k). We define a metric to measure the stability of the network based upon the concept of *giant component*. Giant component signifies the largest connected component in the network whose size is of the order of size of the network [116]. The dissolution of the giant component due to the removal of nodes indicates the percolation point and we use percolation threshold as the stability metric.

Based upon these models and metrics, we develop an analytical framework to examine the stability of superpeer networks against node removal. We apply this framework to measure the stability of superpeer networks against peer churn [110]. We estimate the impact of peer churn on the network and report the influence of peer-superpeer degree and their respective fractions on the network stability. We show that superpeer networks exhibit stable behavior against churn which is supported by recent measurement studies [158, 160]. We perform stochastic simulations to validate the theoretical framework developed in this chapter.

The rest of the chapter is organized as follows. In section 3.2, we formalize and model various environmental parameters that will be used throughout the thesis. This includes modeling network topology with the help of degree distribution, various node dynamics like churn and attack through node dynamics. Here we also explain the simulation environment generated to mimic large superpeer networks and specify the methodology to measure the stability of the network. In section 3.3, we develop an analytical framework to analyze the stability of peer-to-peer networks against various node dynamics. In section 3.4, we utilize the developed formalism to assess the stability of superpeer networks against peer churn. We also validate the theoretical results with the help of simulations. Finally section 3.5 concludes the chapter.

3.2 Environment definitions

In this section, we formally model different peer-to-peer networks and churn/attacks to develop the analytical framework. Also we define the stability metric and explain the simulation undertaken to verify the theoretical results.

3.2.1 Modeling superpeer networks

A survey in the literature reveals that most of the real world networks can be represented as large scale complex graphs. In this thesis, we use a wide range of superpeer network representations; from scale free network to real Gnutella network. The different types of superpeer overlay networks can be modeled using uniform framework of probability distribution p_k, where p_k is the probability that a randomly chosen node has degree k. The networks discussed next can be categorized into two segments **1.** networks that are modeled to represent superpeer networks **2.** networks that are simulated to replicate the real world networks. We specifically focus on the generation of Gnutella networks as the representative commercial peer-to-peer networks.

Superpeer networks modeled as complex graphs

We model the superpeer networks with different kind of complex graphs; (a) bimodal network (b) mixed poisson network (c) scale free networks

(a) Bimodal network: We believe bimodal network is simple enough to understand and analyze; at the same time it captures the essence of commercial superpeer networks [133, 139]. In bimodal network, superpeer topology can be modeled by bimodal degree distribution where a large fraction (r) of peer nodes with small degree k_l are connected with superpeers and few superpeer nodes $(1 - r)$ with high degree k_m are connected to each other. Therefore only two separate degrees are allowed in this kind of network. Formally

$$p_k > 0 \quad if \;\; k = k_l, k_m; \quad p_k = 0 \quad otherwise \tag{3.1}$$

k_l & k_m are degrees of peers and superpeers respectively. Therefore $p_{k_l} = r$ and $p_{k_m} = (1 - r)$.

(b) Mixed poisson network: We can model the superpeer networks in a more sophisticated way as mixed poisson network which is the superposition of two Poisson distributions with different average degrees $\langle k \rangle$ [19]. In mixed poisson network, interconnection between superpeers are selected to approximate a E-R graph [48, 49] which follows Poisson distribution. Similarly, the degree distribution of peers follows another Poisson distribution. The average degree of the superpeers is much higher than peers. Mathematically, if r be the fraction of peers in the network[1] and rest are superpeers then degree distribution of the network

$$p_k = rp_{k_{pr}} + (1-r)p_{k_{spr}} \tag{3.2}$$

where degree distribution of peers $p_{k_{pr}} = \frac{\langle k_p \rangle^{k_{pr}} e^{-\langle k_p \rangle}}{k_{pr}!}$ and superpeers $p_{k_{spr}} = \frac{\langle k_{sp} \rangle^{k_{spr}} e^{-\langle k_{sp} \rangle}}{k_{spr}!}$ follow Poisson distribution with average degree $\langle k_p \rangle$ and $\langle k_{sp} \rangle$ respectively, and $\langle k_p \rangle << \langle k_{sp} \rangle$. The average degree of the mixed poisson network becomes

$$\langle k \rangle = r\langle k_p \rangle + (1-r)\langle k_{sp} \rangle \tag{3.3}$$

(c) Scale free networks: Communication networks have been intensively studied during the last several years. It has been found that many peer-to-peer networks may be characterized by the power law degree distribution

$$p_k \sim k^{-\alpha} \tag{3.4}$$

where p_k is the fraction of nodes in the network of degree k. Here exponent α is a constant whose value is typically in the range $2 < \alpha < 3$.

Superpeer networks simulated as Gnutella networks

We simulate commercial Gnutella networks following two different strategies (a) bootstrapping protocols (b) topological snapshot. We explain them one after another.

(a) Gnutella network generated from bootstrapping protocol: In order to simulate the Gnutella network, we follow the procedure described in [82]. The procedure is based upon the bootstrapping protocol followed by different commercial Gnutella clients like limewire, mutella, etc. In order to join the network, peers execute

[1] If total number of nodes in the network is N and out of them n_p is the number of peers then $r = \frac{n_p}{N}$.

these bootstrapping protocols in which they discover other online peers/superpeers and establish connections with them. Gnutella web caching system is used to determine the addresses of online peers/superpeers that would allow the incoming peer to join the network. We simulate the bootstrapping protocol as the user level thread which is executed by the incoming peer and construct the emerging network. We find that as the number of nodes in the networks (N) grows towards a very large value ($N \rightarrow \infty$), the topology stabilizes to some specific degree distribution.

(b) Gnutella network generated from the topological snapshot: In addition to simulating the Gnutella network following the bootstrapping protocol, we simulate another Gnutella network following the snapshots obtained from the Multimedia & Internetworking Research Group of University of Oregon, USA [1]. The snapshot is obtained by the research group during September 2004 and the size of the network simulated from the snapshot is of $1, 31, 869$ nodes.

3.2.2 Churn and attack models

Nodes in the p2p network join and leave the network randomly without any central coordination. This churn of nodes might partition the network into smaller fragments and breakdown communication among peers [146]. In addition, stability of the overlay network can get severely affected through intended attacks targeted towards the important peers [138, 145]. The importance of a node is mainly characterized by its connectivity and bandwidth. We model peer churn and attacks as the removal of nodes from the network along with their adjacent links. In our framework f_k is used to specify the churn and attack models where f_k be the probability of removal of a node of degree k. Peer churn can be modeled by two different kinds of node failures in the network, namely degree independent failure and degree dependent failure. The formal modeling of the churn is presented next.

- In **degree independent failure** (or random failure), nodes are randomly selected and removed from the network. Hence the probability of removal of any node is constant, degree independent and equal for all other nodes in the graph.

Mathematically, probability of removal a k degree node after this kind of failure is $f_k = f$ (independent of k).

- In superpeer networks, peers having higher connectivity (e.g. superpeers) are more stable in the network than the peers having lower connectivity (e.g. connected through dial up line) because those loosely connected peers enter and leave the network quite frequently. These observations lead us to model *degree dependent failure.*

  In **degree dependent failure**, probability of failure of a node (f_k) having degree k is inversely proportional to k^γ. i.e $f_k \propto 1/k^\gamma \Rightarrow f_k = \alpha/k^\gamma$ where $0 \leq \alpha \leq 1$ and γ is a real number. Therefore probability of the presence of a node having degree k after this kind of failure is $q_k = (1 - \frac{\alpha}{k^\gamma})$.

During attack, attackers remove the highest degree nodes from the network. To identify the highest degree nodes in the network, the availability of the information to the attacker regarding the topology of the network is important. Based upon the availability of the information, we define two kinds of attacks, namely deterministic attack and degree dependent attack. The formal modeling of attacks are presented next.

- In **deterministic attack**, the nodes having high degrees are progressively removed. Formally

  1. $f_k = 1$ when $k > k_{max}$

     $0 \leq f_k < 1$ when $k = k_{max}$

  2. $f_k = 0$ when $k < k_{max}$. This removes all the nodes from the network with degree greater than k_{max} and a fraction of nodes having degree equal to k_{max}.

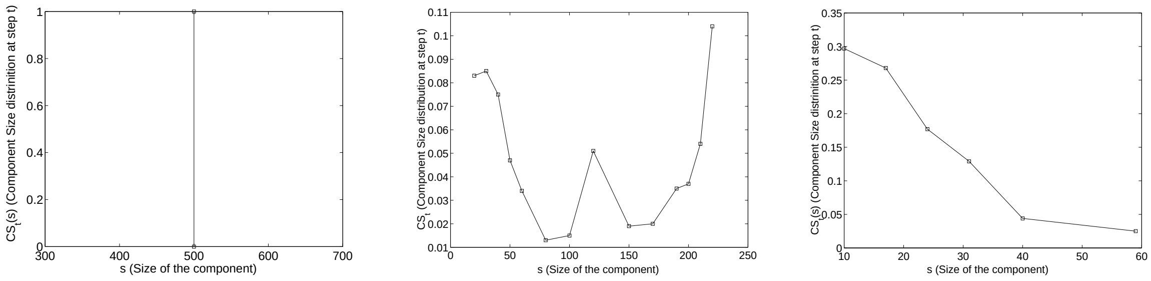
- In **degree dependent attack**, the probability of removal of a node (f_k) having degree k is proportional to k^γ. i.e $f_k \propto k^\gamma \Rightarrow f_k = Ck^\gamma$ where γ is a real number. γ is associated with the degree of knowledge of the attacker about the topological structure of the network. A high γ indicates the availability of sufficient amount of information to the attacker which eventually concentrates attacking a few high degree nodes. On the other hand, low γ reduces the network information to the attacker, hence attack in this case is mostly tilted towards random nodes in the network.

This is important to note that degree dependent attack can also reproduce the degree independent failure and degree dependent failure just by adjusting the parameter γ in the range of $0 \leq \gamma < 1$. However, this is not still explicit whether degree dependent attack can also capture the deterministic attack.

3.2.3 Stability metric

The stability of overlay networks is measured in terms of certain fraction of nodes (f_c) called percolation threshold [22,116], removal of which disintegrates the network into large number of small, disconnected components. Below that threshold, there exists a large connected component which spans the entire network. This connected component is also termed as the giant component. The value of percolation threshold f_c theoretically signifies the stability of the network, higher value indicates greater stability against churn and attack. The existence of giant component can be mathematically captured by the ratio $\kappa = \langle k^2 \rangle / \langle k \rangle$ where $\langle k \rangle$ and $\langle k^2 \rangle$ are the first and second moments of the degree distribution; the value of $\kappa \geq 2$ indicates the situation where stability of the network is maintained [28,115]. The minimum fraction of nodes required to be removed to put the value of κ equal to 2 is defined as the percolation threshold f_c.

Calculation of percolation threshold f_c through simulation is a challenging task. Theoretically, the size of the network as well as size of the giant component is infinite. The removal of f_c fraction of nodes reduces the giant component size from infinite to finite. However in practice, all the networks (generated from simulation) are finite in size. Hence, suitable methods need to be developed to simulate that phenomenon in finite graphs. In section 2.2.3 of Chapter 2, we have presented a detailed survey on various stability metrics. In this section, we describe two different techniques to calculate percolation threshold from simulation. The first one developed by us gives more practical insights regarding the change in the component sizes at the



(a) Initial component size distribution

(b) Intermediate component size distribution

(c) Component size distribution at percolation point

Figure 3.1: The above plots represent the change in the component size distribution during percolation process and indicates the percolation threshold. Initially there exists only single giant component of size 500 which disintegrates in subsequent steps.

percolation point. The second one follows a more theoretical approach as specified in [56]. Fig. 3.3 shows that both of these approaches gives reasonably close percolation thresholds during simulation. Henceforth throughout in this thesis, we use the second approach to calculate percolation threshold f_c.

Calculating percolation threshold from simulation

Technique 1.

Here we take cue from condensation theory to develop the metric to measure the percolation threshold experimentally [102, 137]. During the simulation, we remove a fraction of nodes f^t (following attack model f_k) from the network in step t and check whether we reach the percolation point. If not, then in the next step $t+1$ we remove $f^{t+1} = f^t + \epsilon$ fraction of nodes from the network and check again. This process is continued until we reach the percolation point. After each step, we find out the status of the network in terms of the number and size of the components formed. We collect the statistics of s and n_s where s denotes size of the components and n_s, number of components of size s and define the normalized component size distribution $CS_t(s) = sn_s / \sum_s sn_s$ at step t. We compute $CS_t(s)$ for all the steps starting from $t = 1$

and observe the behavior of $CS_t(s)$ after each step (Fig. 3.1). Initially the $CS_t(s)$ shows unimodal character confirming a single connected component (Fig. 3.1(a)) or bimodal character (Fig. 3.1(b)) indicating a large component along with a set of small components. As the fraction of nodes removed from the network increases gradually, the network disintegrates into several components. This leads to the change in the behavior of $CS_t(s)$ whereby at a particular step t_n, $CS_{t_n}(s)$ becomes a monotonically decreasing function indicating t_n as percolation point (Fig. 3.1(c)). Therefore t_n is considered as the time step where percolation occurs and the total fraction of nodes removed at that step f^{t_n} specifies the percolation threshold.

Technique 2

We follow the method used in [56] to find the simulated value of the percolation threshold f_c. In this experiment, we remove an arbitrary fraction of nodes $f_{c_{candidate}}$ (following attack model f_k) from the network following the probability distribution f_k. This $f_{c_{candidate}}$ works as a candidate solution for the percolation threshold f_c. After removal of $f_{c_{candidate}}$, we compute the new degree distribution p'_k, first and second moments $\langle k \rangle$ and $\langle k^2 \rangle$ of p'_k and subsequently calculate $\kappa = \langle k^2 \rangle / \langle k \rangle$. This procedure is repeated for 100 realizations for a candidate $f_{c_{candidate}}$ and the number of times κ becomes greater than 2 is calculated. We take different values of $f_{c_{candidate}}$ ($0 < f_{c_{candidate}} < 1$) to perform the experiment. The particular value of $f_{c_{candidate}}$, for which 50% of times (realizations) $\kappa > 2$ and rest 50% of times $\kappa \leq 2$ is considered to be the simulated percolation threshold f_c.

3.2.4 Simulation environment

In simulation, a peer-to-peer network is represented by a simple undirected graph stored as an adjacency list. In order to generate the topology, every node is assigned a degree according to the specific degree distribution. Thereafter the edges are generated using the "matching method" [106]. Some of the edges are then rewired using "switching method" to generate sufficient randomness in the graph [105]. In our experiment, we simulate the overlay network by generating graphs with 5000 nodes.

Churn or attack on a peer effectively means deletion of the node and its corresponding edges. We implement this phenomenon by removing a fraction of nodes in each step depending on the disrupting event in the network. In the case of churn, nodes are randomly selected using a time-seeded pseudo-random number generator and its edges are removed from the adjacency list. For targeted attack, high degree nodes in the network are removed sequentially in each step until the percolation point is reached. We perform each simulation for 500 times and take the average of the percolation threshold.

3.3 Developing analytical framework using generating function formalism

In this section, we derive an analytical framework for measuring the stability of overlay structures undergoing any kind of disturbances in the network. With the help of this framework, we find the critical condition for break down of the connectivity of the network. This is an extension of Newman et al.'s theory on random graphs [127]. We assume that we have an infinite system, and so before any failure or attack the biggest cluster size in the system is infinite. Theoretically, the question that we want to answer is how severe should be the failure or attack to make the biggest cluster size in the system finite.

We start out by giving some definitions. We have already defined p_k as the probability of finding a randomly chosen node with degree k and f_k as the probability that a node of degree k is removed due to failure or attack. Thus, p_k models the ensemble of overlay structures and f_k models the disruptive events that take place in the network. Correspondingly $q_k = 1 - f_k$ is the probability that a node of degree k survives after node removal process. We are going to establish the relationship between stability and p_k and q_k i.e. $(1 - f_k)$ using the generating function formalism.

Generating function

Generating function has been widely used to model various stochastic processes [22, 127]. A brief introduction of generating function follows.

A generating function $G(x)$ is formally a power series of x which encodes some probability distribution. Let us assume that $G(x)$ generates the degree distribution of the network given by p_k, then the generating function takes the form

$$G(x) = \sum_{k=0}^{\infty} p_k x^k \tag{3.5}$$

The connection between the generating function and the probability distribution it generates is given by

$$p_k = \lim_{x \longrightarrow 0} \frac{1}{k!} \frac{d^k G(x)}{dx^k} \tag{3.6}$$

Another important property of generating functions is that the average of the index of the probability, i.e., for $G(x)$ the average degree z of a vertex, can be expressed simply by

$$z = \langle k \rangle = \sum_{k=0}^{\infty} k p_k = G'(1) \tag{3.7}$$

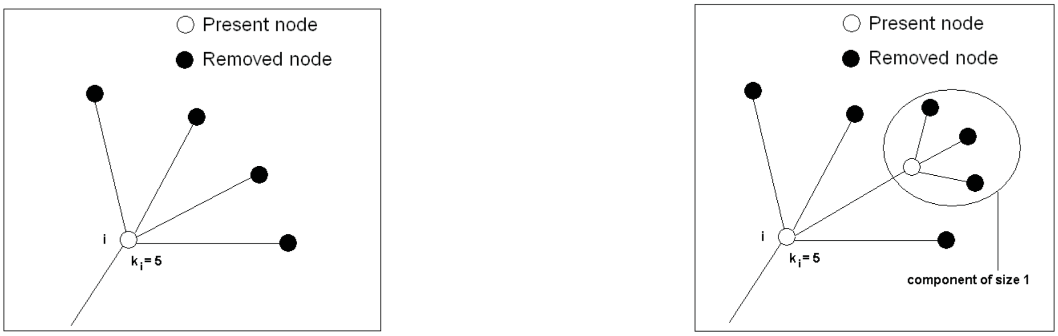
Using this formalism we can formulate the generating function $H_0(x)$ which generates the distribution of the component sizes to which a randomly selected *node* belongs to. Subsequently the average size of the components can be calculated from $H_0'(1)$. When this average component size becomes infinity, it indicates the emergence of giant component and hence we can derive the critical condition for the stability of the giant component. However to formulate $H_0(x)$, we have to use a set of generating functions that are specified below.

Some useful generating functions

- $H_1(x)$ generates the distribution of the component sizes that are reached by choosing a *random edge* and following it to one of its ends.

- $F_1(x)$ generates the probability distribution of the outgoing edges of the *first neighbor* of a randomly chosen *node* after the process of removal of some portion of nodes is completed.

- $F_0(x)$ is the generating function associated with the probability of a *node* having degree k to be present in the network after the disruptive event.



(a) Calculation of s_1

(b) Calculation of s_2

Figure 3.2: Schematic diagram explains the calculation of s_1 and s_2. White node indicates the node reached by following a random edge and black nodes indicate the removed nodes.

Derivation of $F_0(x)$

The generating function $F_0(x)$ specifies the probability of finding a node of degree k to be present in the network after the failure or attack. Since $p_k q_k$ is the probability of finding a node of degree k to be present after the disruptive event, applying the definition of generating function (Eq. 3.5), we find that $F_0(x)$ takes the form

$$F_0(x) = \sum_{k=0}^{\infty} p_k q_k x^k \tag{3.8}$$

Derivation of $F_1(x)$

To reach the first neighbor of a randomly chosen node, we have to pick up one of its outgoing links randomly and follow it until we reach the other end. Hence the probability distribution generated by $F_1(x)$ is same as the probability distribution of the outgoing edges of a node reached by following a random edge. Therefore we derive the generating function $F_1(x)$, with the help of another generating function $A(x)$ which is based upon the probability of finding a randomly chosen edge connected to a node of degree k.

Derivation of $A(x)$:

If we think of an edge connecting two nodes i and j as actually two edges; one going from i to j, and another from j to i, then total number of such edges in the system becomes $\sum_{k=0}^{\infty} kn_k$, where n_k is the number of nodes with degree k in the system, which can be expressed as $n_k = Np_k$ (N being the total number of nodes in the system). The expected number of edges connected to nodes of degree k which remained present after the node removal event is kn_kq_k. So, the probability of finding a randomly chosen edge connected to a node of degree k becomes

$$p_{on}(k) \;=\; \frac{kn_kq_k}{\sum_{k=0}^{\infty} kn_k} = \frac{kp_kq_k}{\sum_{k=0}^{\infty} kp_k} = \frac{kp_kq_k}{z} \tag{3.9}$$

In consequence the generating function associated to the probability $p_{on}(k)$ is

$$A(x) \;=\; \sum_{k=0}^{\infty} p_{on}(k)x^k = \sum_{k=0}^{\infty} \frac{kp_kq_k}{z} x^k$$

Since $\sum_{k=0}^{\infty} kp_kq_kx^k$ can be expressed as $xF_0'(x)$ therefore with the help of Eq. (3.7)

$$A(x) = xF_0'(x)/G'(1) \tag{3.10}$$

Derivation of $F_1(x)$:

The generating function $F_1(x)$ is based upon the probability distribution signifying the outgoing degree of a node reached following a random edge. We know that a node having degree k arrived following a random edge has only $k-1$ outgoing links that leave from that node. Hence probability of finding an existing node (that survives after the disruptive event) of $k-1$ outgoing edges reached following a random edge is $p_{on}(k) = \frac{kp_kq_k}{z}$ as defined in Eq. (3.9). Therefore probability distribution of the outgoing edges of the first neighbor of a randomly chosen node can be generated by

$$F_1(x) = \sum_{k=1}^{\infty} p_{on}x^{k-1} = \sum_{k=1}^{\infty} \frac{kp_kq_k}{z} x^{k-1} = F_0'(x)/z \tag{3.11}$$

Derivation of $H_1(x)$

The function $H_1(x)$ generates the distribution of cluster sizes reached by following an edge chosen uniformly at random. Without loss of generality, we assume that following an edge, we can reach either a non existent node (node removed during deletion) or an existent node. The probability of following the randomly chosen edge and finding an existing/present node of degree zero is zero, the probability of finding an existing node of degree one is p_1q_1/z, the probability of finding an existing node of degree two is $2p_2q_2/z$, and so on. So the probability of finding a node following a random edge is $\sum_{k=0}^{\infty} kp_kq_k/z = F_1(1)$. In consequence, the probability of finding an edge that leads to a node which has been removed is $1-F_1(1)$. Clearly this is also the probability of following a randomly chosen edge that leads to a zero size component. Therefore if s_0 is the coefficient that accompanies x^0 in $H_1(x)$ then $s_0 = 1 - F_1(1)$. To find the full expression of $H_1(x)$ we have still to look for the probabilities that accompany non-zero size components. We find those probabilities next with the help of induction method.

Calculation of s_1, s_2 *etc:* We calculate the probability s_1 of finding a component of size 1 by following a randomly chosen edge. This is nothing else than the sum of the probabilities of following an edge and finding a node of degree k which has its other $k-1$ edges connected to zero size components (all the nodes in these components are removed) (Fig. 5.1(a)). This can be expressed as:

$$\begin{aligned} s_1 &= \sum_{k=1}^{\infty} \frac{kp_kq_k}{z}(1-F_1(1))^{k-1} \\ &= F_1(H_1(0)) = \lim_{x\longrightarrow 0} \frac{1}{1!}\frac{d(s_0 + xF_1(H_1(x)))}{dx} \end{aligned}$$

where $p_{on}(k) = kp_kq_k/z$ and $(1-F_1(1))^{k-1}$ is the probability of taking randomly $k-1$ edges and finding that all of them are attached to zero size components.

Knowing this we can easily calculate s_2, the probability of finding a component of size 2 by following a randomly chosen edge. s_2 is the sum of the probability of following a randomly chosen edge that leads to a node of degree k which is connected to $k-2$ zero size components, and has also an edge that leads to a component of size 1 (Fig. 6.2(a)). This can be expressed as

$$\begin{aligned} s_2 &= \sum_{k=2}^{\infty} \frac{(k-1)kp_kq_k}{z}(1-F_1(1))^{k-2}s_1 \\ &= F_1'(H_1(0))H_1'(0) = \lim_{x\longrightarrow 0} \frac{1}{2!}\frac{d^2(s_0 + xF_1(H_1(x)))}{dx^2} \end{aligned}$$

where $(1-F_1(1))^{k-2}s_1$ is the probability of taking randomly $k-1$ edges and finding that $k-2$ edges are attached to zero size components, and one to a size 1 component. The term $k-1$ in s_2 indicates that there are $k-1$ possible configurations for these edges.

Similarly, we can calculate the probability of finding a component of size 3 by following a randomly chosen edge

$$s_3 = \lim_{x \longrightarrow 0} \frac{1}{3!} \frac{d^3(s_0 + xF_1(H_1(x)))}{dx^3}$$

and so on. This suggests a self-consistence equation for $H_1(x)$ that generates the distribution of component sizes of nodes that are reached by randomly chosen edge after the disruptive event

$$\begin{aligned} H_1(x) &= s_0 + xF_1(H_1(x)) \\ &= 1 - F_1(1) + xF_1(H_1(x)) \end{aligned} \tag{3.12}$$

It can be easily verified that Eq. (3.12) leads to the correct expressions of s_0, s_1,..., s_n by applying Eq. (3.6).

Derivation of $H_0(x)$

Along similar lines we can obtain the generating function $H_0(x)$ of the distribution of the component size to which a randomly chosen node belongs to. The probability that a randomly chosen node belongs to a component of size zero after the disruptive event is $1 - F_0(1)$. Similarly the probability of a randomly chosen node to belong to some nonzero size component depends on the size of the components where all its first neighbors belong to. Hence the expression for $H_0(x)$ takes the form:

$$H_0(x) = (1 - F_0(1)) + xF_0(H_1(x)) \tag{3.13}$$

Finally from Eq. (3.13) and recalling the definition of average given by Eq. (3.7), we can obtain the average size of the components:

$$H_0'(1) = \langle s \rangle = F_0(1) + \frac{F_0'(1)F_1(1)}{1 - F_1'(1)} \tag{3.14}$$

As mentioned above, we are interested in knowing the threshold at which the average cluster size becomes infinite. Clearly Eq. (3.14) diverges when $1 - F_1'(1) = 0 \Rightarrow F_1'(1) = 1$, and this critical condition sets the threshold between finite and infinite cluster sizes. We present an intuitive explanation for this critical condition of giant component disruption. $F_1'(1)$ represents the average outgoing links of the first neighbor of a randomly chosen node. After the node removal process, if this average number of outgoing links is more than one, then the network should percolate, i.e. it is possible to find an infinite cluster of connected nodes. But if it is less than one, then it is very likely that by following a random edge, we land in a node that has no outgoing link and thus no chance of reaching another existing node.

Finally replacing $F_1'(1)$ by its definition (Eq. (3.11)), we obtain a critical condition for giant component formation

$$\sum_{k=0}^{\infty} kp_k(kq_k - q_k - 1) = 0 \tag{3.15}$$

The significance of the Eq. (3.15) lies in the fact that it states the critical condition for the stability of giant component with respect to any type of graphs (characterized by p_k) undergoing any type of failure or attack (characterized by q_k). Formulating this general formula is one of the primary contributions of this chapter and the thesis [112, 114].

Using the formalism developed, we investigate the stability situation of various superpeer networks elaborated next.

3.4 Stability of superpeer networks against churn

The p2p networks mostly experience churn of peers which we model as the removal of nodes in complex graph. In section 3.2.2, we model the peer churn by two kinds of node failures - degree independent and degree dependent. In the next two subsections, we deal with these two kinds of failures and investigate their effect on the stability of superpeer networks.

3.4.1 Stability analysis against degree independent failure

In this section, we discuss the effect of degree independent failure in generalized random graph. If $q = q_r$ is the critical fraction of nodes whose presence in the graph is essential for the stability of the giant component after this kind of failure then according to Eq. (3.15)

$$\sum_{k=0}^{\infty} kp_k(kq_r - q_r - 1) = 0$$

$$\Rightarrow q_r = \frac{1}{\frac{\langle k^2 \rangle}{\langle k \rangle} - 1} \tag{3.16}$$

Now if f_r is the critical fraction of nodes whose random removal disintegrates the giant component then $f_r = 1 - q_r$. Therefore percolation threshold

$$f_r = 1 - \frac{1}{\frac{\langle k^2 \rangle}{\langle k \rangle} - 1} \tag{3.17}$$

This is the well known condition [28] (derived differently) for the disappearance of the giant component due to random failure. This shows that the proposed general formula (Eq. (3.15)) can be used to reproduce Eq. (3.17) as a special condition.

3.4.2 Superpeer networks against degree independent failure

The superpeer networks mostly experience the churn of peers which can be modeled by the failure of nodes in complex graph. In this section, we use our equations to show that stability of the superpeer networks is quite unaffected due to churn of peers. This observation is consistent with results from real life experiment [146, 158, 160]. We investigate the change of percolation threshold (f_c) due to the change of fraction of peers (r) and the connectivity of the superpeers (k_m) in the networks for various types of failures. To ensure fair comparisons, we keep *the average degree* $\langle k \rangle$ *constant for all graphs.* We verify our theoretical results with the help of simulation. First, we consider the bimodal networks for our analysis. Subsequently, we analyze the more realistic model of mixed poisson networks.

Bimodal Networks

The bimodal degree distribution is modeled in the following way. Let r be the fraction of peers in the networks having degree k_l and rest are superpeers having degree k_m

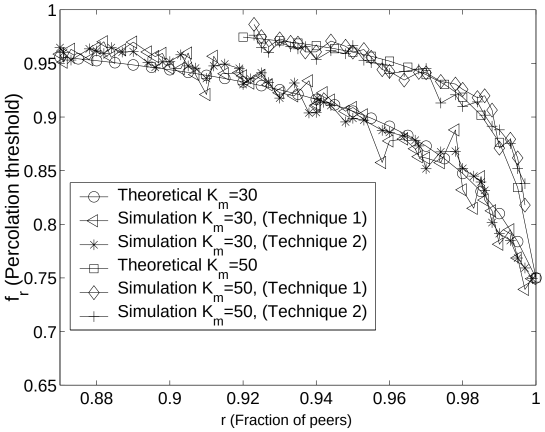


Figure 3.3: The above plots represent a comparative study of theoretical and simulation results of stability for two bimodal networks undergoing churn. Here X-axis represents the fraction of peer nodes (r) existing in the network and Y-axis represents the corresponding percolation threshold (f_r). We keep the average degree $\langle k \rangle = 5$ fixed and vary the superpeer degree $k_m = 30, 50$ for two plots. The tangential line indicates the change in peer degree due to change in the peer fraction r.

where $k_l << k_m$ that is in bimodal degree distribution, p_k becomes non zero only at k_l and k_m (Eq. (3.1)). Mathematically $k_l p_{k_l} + k_m p_{k_m} = \langle k \rangle$ and $p_{k_l} + p_{k_m} = 1$ which provides

$$p_{k_m} = \frac{\langle k \rangle - k_l}{k_m - k_l} \qquad\qquad p_{k_l} = \frac{k_m - \langle k \rangle}{k_m - k_l} \tag{3.18}$$

Degree independent failure

Therefore second moment of the degree distribution is given as $\langle k^2 \rangle = k_m^2 p_{k_m} + k_l^2 p_{k_l} = \langle k \rangle (k_l + k_m) - k_l k_m$. Consequently, using Eq. (3.17) for random failure, we get

$$f_r = 1 - \frac{\langle k \rangle}{\langle k \rangle (k_l + k_m - 1) - k_l k_m} \tag{3.19}$$

The equation can be written in terms of the fraction of peers. Peer degree in the network denoted as k_l can be derived as $\frac{\langle k \rangle - (1-r)k_m}{r}$, where r is fraction of peers.

Hence percolation threshold

$$f_r = 1 - \frac{\langle k \rangle r}{\langle k \rangle^2 - 2\langle k \rangle k_m + 2rk_m\langle k \rangle - r\langle k \rangle + k_m^2 - rk_m^2} \tag{3.20}$$

Feasible fraction of peers : Since the peer degree k_l needs to be at least one ($k_l = 1$) to be connected in the network therefore

$$k_l = \frac{\langle k \rangle - (1 - r_c)k_m}{r_c} \geq 1 \tag{3.21}$$

$$\Rightarrow r_c \geq \frac{k_m - \langle k \rangle}{k_m - 1} \tag{3.22}$$

That means we can form a bimodal network with prescribed peer and superpeer degrees only if the fraction of peers is greater than the feasible peer fraction (r_c). For $k_m = 30, 50$ this feasible fraction r_c becomes $0.862, 0.918$ respectively. Below that fraction, there does not exist any network, therefore our theoretical analysis as well as simulations are performed with fraction r above the feasible fraction r_c.

Using Eq. (3.20), we study the variation of percolation threshold (f_r) due to the change in the fraction of peers (r) for networks with two different superpeer degrees and compare the results through simulation (Fig. 3.3). In order to obtain percolation threshold during simulation, we use the two techniques as described in section 3.2.3. Since both techniques give reasonably close percolation thresholds during simulation, we use 'technique 2' to calculate f_c throughout this thesis. *It can be observed from Fig. 3.3 that simulation results match closely with theoretical predictions which shows the success of our theoretical framework.*

Observations

1. It is important to observe that for the entire range of peer fractions, the percolation threshold f_r is greater than 0.7 which implies that superpeer networks are quite robust against churn. Since churn affects peers and superpeers depending upon their individual fraction in the network, peers are affected much more than superpeers. The removal of a significant number of low degree peers along with a few high degree superpeers have little impact upon the stability of

the networks. Practical experience also ensures that superpeer networks exhibit high robustness in face of churn [146,158,160].

2. Lower fraction of superpeers in the network (specifically when it is below 5%) results in a sharp fall of f_r, that is the vulnerability of the network drastically increases when the fraction of superpeers is below 5%. When the fraction of superpeers are high, the constituent peers are only connected to superpeers (and not within themselves), hence stability of the network depends entirely upon superpeers. As fraction of superpeer reduces below 5%, peer degree becomes quite high (4 to 5). This gives rise to situations where significant number of peers are not connected directly to the superpeers, but connected within themselves. Hence removal of only a few randomly selected peers can also result in the removal of fellow peers. This produces an avalanche effect which results in a drastic reduction of stability of the network in this region.

Effect of superpeer degree: It is seen from Fig. 3.3 that increase of superpeer degree k_m also increases the stability of the network (any vertical line in the plot for a given fraction of peers r). Although the ratio $\frac{superpeerfraction}{peerfrcation}$ is constant in both cases, the higher (lower) superpeer (peer) degree leads to the higher participation of superpeers. In order to have a more fair comparison, next we define a metric namely 'peer contribution' formed using degree of a node and its fraction.

Impact of peer contribution

The peer contribution signifies the fraction of total bandwidth contributed by the peers which essentially determines the amount of influence superpeer nodes exert on the network. Suppose the total bandwidth contributed by peer nodes is X while that by superpeer nodes is Y, then peer contribution becomes $\frac{X}{X+Y}$. In our thesis, we define peer contribution Pr_C by two parameters - peer degree and fraction of peers in the network. Hence $Pr_C = \frac{rk_l}{\langle k \rangle}$ where $\langle k \rangle = rk_l + (1-r)k_m$. For a particular peer contribution Pr_C, the required fraction of peers becomes $r = (1 - \frac{(1-Pr_C)\langle k \rangle}{k_m})$. The percolation threshold f_r is calculated by substituting the peer fraction r in Eq. (3.20) for individual Pr_C which results

$$f_r = 1 - \frac{k_m - (1-Pr_C)k}{kk_m - 2(1-Pr_C)kk_m - k_m + (1-Pr_C)k + km^2(1-Pr_C)} \tag{3.23}$$

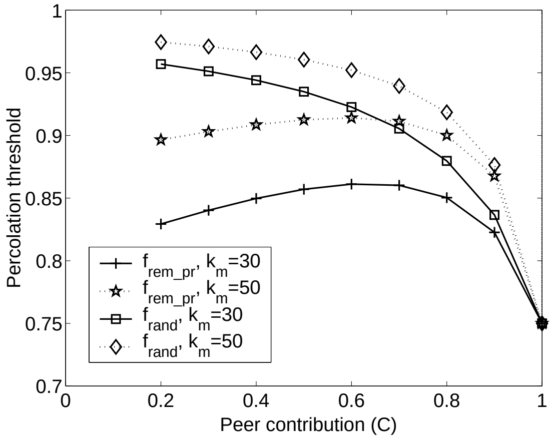


Figure 3.4: The plot represents the impact of peer contribution Pr_C upon the stability of the network against churn. Two different superpeer degrees $k_m = 30, 50$ are considered. f_{rem_pr} represents the fraction of pure peers required to be removed to dissolved the network and f_r indicates the corresponding percolation threshold.

The fraction of peers and superpeers required to be removed for random failure is proportional to their respective share in the network (i.e. $f_{rem\_pr} = rf_r$, $f_{rem\_sp} = (1-r)f_r$). These estimated values of f_{rem_pr}, f_r are plotted with respect to the peer contribution Pr_C (Fig. 3.4) for superpeer degree $k_m = 30, 50$ and average degree $\langle k \rangle = 5$. The theoretical model is sufficient for analysis as the model has been already validated through simulation.

Observations

1. It is interesting to note that increase in peer contribution initially increases the fraction of peers required to be removed (f_{rem_pr}) but after a critical peer contribution ($Pr_{C_{rand}}$), f_{rem_pr} decreases. The fraction of peers required to be removed can be written as $f_{rem\_pr} = rf_r$. Hence the maximum peers required to be removed can be obtained by

$$\frac{df_{rem\_pr}}{dPr_C} = 0 \tag{3.24}$$

$$\Rightarrow \frac{L_1 + Pr_C L_2}{(L_3 + Pr_C L_4)^2} + \left(\frac{L_5 + Pr_C L_6}{L_3 + Pr_C L_4} \right) = 0 \tag{3.25}$$

where L_1, L_2, L_3, L_4, L_5 and L_6 are the constants dependent on superpeer degree k_m and average degree $\langle k\rangle$. By solving Eq. (3.25), we obtain the critical peer contribution $Pr_{C rand}$ maximizing the f_{rem_pr}. Substituting $Pr_C = Pr_{C rand}$ in $\frac{d^2 f_{rem\_pr}}{d^2 Pr_C}$ produces negative value which confirms the maximality of f_{rem_pr}. On solving the Eq. (3.25) for $k_m = 30$ and 50 with average degree $\langle k\rangle = 5$ we get $Pr_{C rand} = 0.63$ and 0.59 respectively.

2. The increase in superpeer degree increases the stability of the network even if peer contribution stays same for both cases. This can be explained with the help of the Eq. (3.23). Let $f_r(k_m)$ and $f_r(k_m + x)$ (x is a positive integer) be the percolation threshold for networks with superpeer degree k_m and $k_m + x$ respectively. As $x > 0$, assuming $k_m >> \langle k\rangle$, we get $f_r(k_m + x) > f_r(k_m)$. Hence increase in superpeer degree increases stability of the network.

Mixed Poisson Networks

In mixed poisson network, let r be the fraction of peers in the network and rest be superpeers [108]. Superpeer nodes are connected to each other to form an E-R network [48,49] with average degree $\langle k_{sp}\rangle$. Similarly peers connected with superpeers form another E-R graph with an average degree $\langle k_p\rangle$ where $\langle k_p\rangle << \langle k_{sp}\rangle$. Now we examine the stability of this kind of superpeer network undergoing churn. In mixed poisson network, first and second moment of the degree distribution become $\langle k\rangle = r\langle k_p\rangle + (1-r)\langle k_{sp}\rangle$ and $\langle k^2\rangle = r\langle k_p^2\rangle + (1-r)\langle k_{sp}^2\rangle$ respectively. If k is a random variable following Poisson distribution then it can be shown that $\langle k^2\rangle \approx \langle k\rangle^2 + \langle k\rangle$. Hence according to Eq. (3.17), percolation threshold becomes

$$f_r = 1 - \frac{r\langle k_p\rangle + (1-r)\langle k_{sp}\rangle}{r\langle k_p\rangle^2 + (1-r)\langle k_{sp}\rangle^2} \tag{3.26}$$

Substituting for $\langle k_p\rangle$ from Eq. (3.3), we get

$$f_r = 1 - \frac{\langle k\rangle r}{\langle k\rangle^2 - 2\langle k\rangle(1-r)\langle k_{sp}\rangle + (1-r)^2\langle k_{sp}\rangle^2 + r(1-r)\langle k_{sp}\rangle^2} \tag{3.27}$$

Similar to bimodal network, in mixed poisson network also, we calculate the feasible fraction of peers r_c

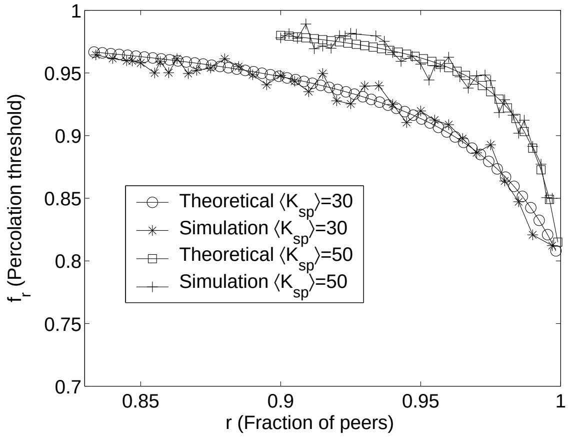


Figure 3.5: The above plots represent a comparative study of theoretical and simulation results of stability for two mixed poisson networks undergoing churn. Here X-axis represents the fraction of peer nodes (r) in the network and Y-axis represents the corresponding percolation threshold (f_r). We keep the average degree $\langle k\rangle = 5$ fixed and vary the mean superpeer degree $\langle k_{sp}\rangle = 30, 50$ for two plots.

Feasible fraction of peers : Since the mean peer degree $\langle k_p\rangle$ needs to be > 0 to be connected in the network therefore

$$\frac{\langle k\rangle - (1 - r_c)\langle k_{sp}\rangle}{r_c} > 0 \tag{3.28}$$

$$\Rightarrow r_c > 1 - \frac{\langle k\rangle}{\langle k_{sp}\rangle} \tag{3.29}$$

That means we can form a connected superpeer network with prescribed peer and superpeer degrees only if the fraction of peers in the network is greater than the feasible peer fraction (r_c). For $\langle k_{sp}\rangle = 30, 50$ this feasible fraction r_r becomes $0.833, 0.90$ respectively. Below that fraction, there does not exist any network, therefore our theoretical analysis as well as simulations are performed with peer fraction r above the feasible fraction r_c.

Using Eq. (3.27), we study the variation of percolation threshold (f_r) due to the change in the fraction of peers (r). We validate the analytically derived result with the help of simulation. We perform the simulation on two mixed poisson networks with average superpeer degree $\langle k_{sp}\rangle = 30$ and 50, keeping the average degree $\langle k\rangle = 5$. Comparative study reveals that networks having higher superpeer degree exhibit more robustness than with lower superpeer degree for any peer-superpeer ratio. *It can be observed from Fig. 3.5 that simulation results match closely with theoretical predictions which shows the success of our theoretical framework.*

The comparative study of the bimodal network and mixed poisson network reveals that both of them exhibit similar behavior in face of degree independent failure. Hence, in the next section of degree dependent failure, we take bimodal network as the representative topology.

3.4.3 Stability analysis against degree dependent failure

In p2p networks, the peers (or superpeers) having higher connectivity are much more stable and reliable than the nodes having lower connectivity. Therefore, probability of the presence of a node having degree k after this kind of failure is

$$q_k = (1 - \frac{\alpha}{k^\gamma}) \tag{3.30}$$

Using equations (3.15) and (3.30), we obtain the following critical condition for the stability of giant component after degree dependent breakdown

$$\langle k^2 \rangle - \alpha \langle k^{2-\gamma} \rangle + \alpha \langle k^{1-\gamma} \rangle - 2\langle k \rangle = 0 \tag{3.31}$$

where percolation threshold is

$$f_d = \sum_{k=0}^{\infty} \frac{\alpha}{k^\gamma} p_k \tag{3.32}$$

Considering the value of $\alpha = 1$, where the fraction of nodes removed due to this kind of failure becomes maximum, the condition for percolation becomes

$$\langle k^{2-\gamma} \rangle - \langle k^{1-\gamma} \rangle = \langle k^2 \rangle - 2\langle k \rangle \tag{3.33}$$

Thus the critical fraction of nodes removed is given by

$$f_d = \sum_{k=0}^{\infty} \frac{1}{k^\gamma} p_k \tag{3.34}$$

where γ satisfies the Eq. (3.33). Thus from the Eq. (3.33) and (3.34), we can determine the variation of percolation threshold f_d for various networks due to degree dependent failure. In the next section, we apply this formalism for superpeer networks and compare with simulation results.

3.4.4 Superpeer networks against degree dependent failure

In bimodal network, r is the fraction of peers in the network having degree k_l and rest are superpeers having degree k_m where $k_l << k_m$. In degree dependent failure, the network percolates if

$$\langle k^{2-\gamma}\rangle - \langle k^{1-\gamma}\rangle = \langle k^2\rangle - 2\langle k\rangle \tag{3.35}$$

If the value of $\gamma = \gamma_c$ satisfies this equation then removal of $f_d = \sum_{k=0}^{\infty} \frac{1}{k^{\gamma_c}} p_k$ fraction of nodes destroys the giant component; however for $\gamma > \gamma_c$, the network survives after node removal. In most of the commercial superpeer networks like KaZaA [83], peers are only directly connected to the local superpeer making their degree $k_l = 1$. In that case, the value of γ_c which percolates the bimodal network can be derived from Eq. (3.35) as

$$\gamma_c = 1 - \frac{\ln \frac{\langle k\rangle(k_m+1) - k_m - 2\langle k\rangle}{\langle k\rangle - 1}}{\ln k_m} \tag{3.36}$$

We plot the variation of the γ_c and percolation threshold f_d with respect to the superpeer degree k_m for various average degree $\langle k\rangle$(Fig 3.6). It is important to notice that the increase in the superpeer degree k_m increases peer fraction r to keep the average degree $\langle k\rangle$ fixed. However here we are interested to understand the impact of superpeer degree upon stability of the networks. It can be observed from Fig. 3.6 that simulation result matches closely with theoretical prediction which shows the success of our theoretical framework.

Observations

1. It can be easily identified from Fig 3.6, that with the increase of superpeer degree k_m, the value of γ_c that percolates the network decreases. This increases the necessary fraction of superpeers required to be removed to breakdown the

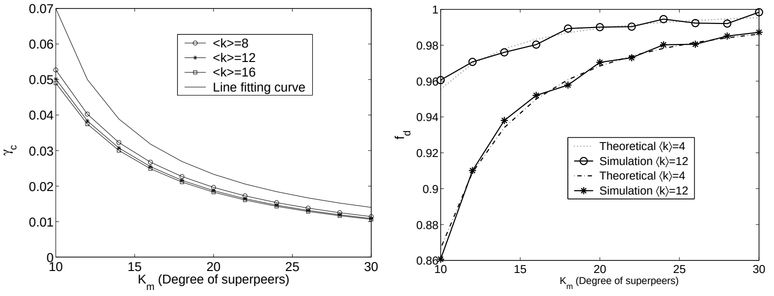


Figure 3.6: Change of γ_c and percolation threshold f_d with respect of superpeer degree k_m for superpeer networks undergoing degree dependent failure. Here mean degree $\langle k \rangle$ varies from 8 to 16. X-axis represents the superpeer degree(k_m) and Y-axis represents the corresponding γ_c and f_d.

network. The nature of γ_c can be approximated by the polynomial $a/(x-b)$ ($0 < a < 1$ and b is some positive integer). Thus the decrease of γ_c follows hyperbolic curve. Since the increase of k_m increases the fraction of peers r, the removal of most of the low degree peers along with a fraction of superpeers increases the percolation threshold f_d.

2. It is interesting to observe that the percolating γ_c remains quite low and less than 0.1 for the entire range of k_m. The reason is that, at smaller values of γ_c, the likelihood that a higher fraction of superpeer nodes would be removed is high. As γ becomes > 0, mainly the lower degree nodes are removed, which are not so useful to break the network down.

3. Another interesting observation is after a certain threshold k_m, the curves become parallel to the X-axis and never cut it thus the value of γ_c is small but never becomes 0 (in that case $f_d = \sum_{k=0}^{\infty} \frac{1}{k^0} p_k = 1$). This implies that for any large value of k_m, although f_d becomes significantly large, however it is required to remove only a part of nodes (and not 'all' the nodes) from the network to dissolve the giant component.

3.5 Conclusion

The basic contributions of this chapter are two folds; first of all, modeling and formalizing various environmental parameters that will be used throughout the thesis and secondly, development of an analytical framework to analyze the stability of various p2p networks against peer churn. We model the peer-to-peer network with the help of probability distribution as well as simulated Gnutella networks from real data and protocols. In addition, we model peer churn and attacks with the help of various node removal techniques. We define percolation threshold as the stability metric and illustrate the procedure to calculate this during simulation.

There have been several interesting observations also which need to be summarized. The analytical framework as well as simulation results show that superpeer networks remain robust under user churn. However, when the fraction of superpeers in the network is less than 5%, the stability of the network sharply decreases for degree independent failure. This result points to a zone where superpeer networks are most vulnerable. Similarly, for degree dependent failure, our analysis shows that increase of superpeer degree improves the stability of the network and the improvement follows a hyperbolic curve. We introduce a new structural metric called 'peer contribution' for more fair analysis and examined its effect upon the stability of the network.

This chapter mainly focuses on the analysis of peer churn on the stability of the superpeer networks. However, in the next chapter, we perform a comprehensive analysis on the impact of *attacks* on the stability of superpeer networks. We develop another theoretical framework to calculate the degree distribution of the deformed network after removal of a fraction of nodes along with their adjacent links. Thereafter the degree distribution of the deformed network is utilized to derive the critical condition for the stability of the network. We show that the method developed in the next chapter is more generalized so that it is able to characterize the impact of various real network issues (like network size, degree-degree correlation etc) on the stability of the network.

Chapter 4

Attack and stability of superpeer networks

In the previous chapter, we have reported the impact of peer churn on superpeer network with the help of an analytical framework. In this chapter, we propose another analytical framework to understand the impact of different types of attacks on superpeer networks. From Chapter 4, we can calculate the stability of uncorrelated large graphs in the same fashion as the previous, however this is more sophisticated than the framework of Chapter 3 in different aspects.

1. In addition to the stability of overall network, the framework of this chapter gives more insights regarding the topology of the network. For instance, the removal of nodes along with their adjacent edges changes the topology of the network. The degree distribution of this deformed network after attack can be calculated with the help of this framework.

2. There are many results that have been derived for infinite networks (similar to previous framework), however, little is known about the stability of finite size networks. The framework developed in this chapter sheds some light on finite size network by proposing an alternative expression for the percolation threshold.

3. Most of the real world networks like Gnutella exhibit degree-degree correlation in the topological structure. Hence understanding the stability of these networks needs to include degree-degree correlation in the calculation (which was not possible in the previous framework). We show that, a little modification of the current framework makes it suitable for the analysis of correlated networks also.

The chapter is organized in the following way. In section 4.1, we develop the analytical framework for stability analysis. In section 4.2 we use the framework to analyze the stability of superpeer networks in face of degree independent attack as well as degree dependent attack modeled in Chapter 3. We show that the degree dependent attack can be used as an unified attack model as other node disturbances may be reproduced by regulating some parameter [109]. We validate our theoretical framework with the help of stochastic simulation. The validation is done in two ways depending upon the generation of superpeer networks, as illustrated in Chapter 3. We start with simple models of superpeer networks, namely bimodal network and mixed poisson network which are simple enough to understand and analyze while at the same time they capture the essential features of the superpeer networks (section 4.2.2). Our framework unfolds various issues such as (i) the available knowledge regarding the topology that helps attackers to breakdown the network (section 4.2.3) (ii) the effect of finiteness of network size on the network stability (section 4.2.4). Afterwards we implement the attack dynamics on the commercial peer-to-peer networks namely Gnutella (section 4.3). Gnutella network is simulated both from the bootstrapping protocol followed by the different Gnutella clients like limewire, mutella etc [82] and from the topological snapshots obtained from [1]. We identify some deviations between theoretical and simulation results due to the presence of degree-degree correlation in Gnutella network. In section 4.4, we further refine our framework to include the degree-degree correlation factor and show that the modified theoretical model gives good agreement with simulated results.

4.1 Development of the analytical framework

In this section, we present the detail derivation of the critical condition for measuring the stability of peer to peer networks undergoing any kinds of attacks [107]. We start out by repeating some definitions mentioned before. Let p_k be the probability of finding a node chosen uniformly at random with degree k. Let f_k be the probability that a node of degree k is removed after the attack. Correspondingly $1 - f_k$ is the probability that a node of degree k survives the attack. In our framework, degree distribution p_k models the ensemble of p2p topologies and f_k models the disruptive events that take place in the network. We are going to establish the relationship between stability, p_k and f_k. This is done as a two step process; in the first step, we calculate the degree distribution of the deformed network after attack. Subsequently in the second step, we use this expression to derive the critical condition of stability of p2p networks against attack.

4.1.1 Deformed topology after attack

In this subsection, we theoretically compute the degree distribution of the deformed topology p'_k after performing an attack on the p2p network of size N with initial degree distribution p_k. The attack in the network can be thought of in the following way. The first step in the attack is to select the nodes that are going to be removed according to the probability distribution f_k. After the selection of the nodes, we divide the network into two subsets, one subset contains the surviving nodes (S) while the other subset comprises of the nodes that are going to be removed (R). This is illustrated in Fig. 4.1. The degree distribution of the surviving subset S is $(1 - f_k)p_k$ while the subset of nodes to be removed R (that is the edges connecting set S and set R) still exist. However, when these nodes are actually removed, the degree distribution of the surviving nodes S is changed due to the removal of the E edges that run between these two subsets.

To calculate the degree distribution after the attack, we have to estimate E. The

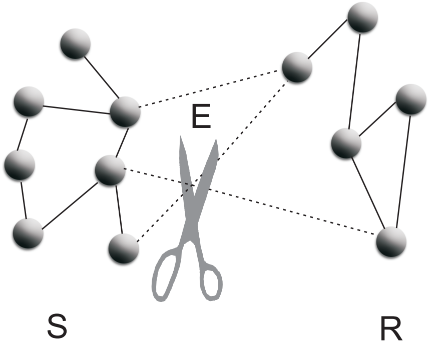


Figure 4.1: The scheme illustrates an attack as consisting of two steps: selection of nodes to be removed (set of removed nodes, R), and cutting of the edges E that run from the surviving nodes (set of surviving nodes, S) to the set of removed nodes R. As the scheme shows, the attack affects the degree of the surviving nodes.

total number of edge tips[1] in the surviving subset S including E links that are going to be removed can be expressed by the sum $\sum_{j=0}^{\infty} j\, n_j\, (1-f_j)$ where $n_j = Np_j$ is the total number of nodes in the network having degree j. Now kn_kf_k gives the total number of edge tips connected with all the k degree nodes in the removed subset R. Therefore $\sum_k kn_kf_k$ becomes the total number of tips in R. Hence the probability of a randomly chosen tip of an edge to be removed becomes $\frac{\sum_k kn_kf_k}{\sum_k kn_k}$. Subsequently the probability of a randomly chosen tip of an edge to be removed (i.e. member of set R) and another tip of that edge being connected to either set S or R becomes $\frac{\sum_k kn_kf_k}{\sum_k kn_k - 1}$ (since a tip cannot be connected to itself). As the network is uncorrelated, it is equally probable that the other end of the removed tip (member of set R) is connected to the nodes of set S or set R. Assuming this unbiasness, the total number of edge tips in set R connected to the nodes of the set S can be expressed as

$$E = \left(\frac{\sum_{i=0}^{\infty} i\, n_i\, f_i}{(\sum_{k=0}^{\infty} k\, n_k) - 1} \right) \sum_{j=0}^{\infty} j\, n_j\, (1-f_j) \tag{4.1}$$

Knowing this, the probability ϕ of finding an edge in the surviving subset S, that is

[1] We assume that each edge consists of two end tips. Hence the total number of tips in the network is twice the number of edges.

connected to a node of the other subset R can be expressed as

$$\phi = \frac{E}{\sum_{i=0}^{\infty} i\, n_i\, (1 - f_i)} = \frac{E}{N \sum_{i=0}^{\infty} i\, p_i\, (1 - f_i)} = \frac{\sum_{i=0}^{\infty} i\ p_i\, f_i}{(\sum_{k=0}^{\infty} k\, p_k) - 1/N}\,. \tag{4.2}$$

In large scale networks, $\lim_{N\to\infty} \phi = \frac{\sum_{i=0}^{\infty} i\ p_i\, f_i}{\sum_{k=0}^{\infty} k\, p_k}$

The probability p^s_q of finding a node with degree q in the surviving subset S (before cutting the E edges) simply becomes

$$p^s_q = \frac{(1 - f_q) p_q}{1 - \sum_{i=0}^{\infty} p_i f_i}\,. \tag{4.3}$$

The removal of nodes can only lead to a decrease in the degree of a survived node. If we find a node of degree k that has survived, it can be due to the fact that originally its degree was $k + q$ and k of its edges survived while q (q may be zero also) got removed. For example, the fraction of nodes having degree k after attack i.e. p'_k is given by the fraction of p^s_k nodes, who did not lose any link, and a fraction of p^s_{k+1} nodes who lost one link but rest k links survived, a fraction of p^s_{k+2} nodes who lost two links but rest k links survived and so on. Hence using the concept of binomial distribution and from the equations (4.2) and (4.3), we obtain the following expression for p'_k:

$$p'_k = \sum_{q=k}^{\infty} \begin{pmatrix} q \\ k \end{pmatrix} \phi^{q-k} (1 - \phi)^k\, p^s_q\,. \tag{4.4}$$

Eq. (4.4) can be iteratively evaluated by replacing p_k with p'_k into Eqs. (4.1) to (4.4).

4.1.2 Critical condition for stability

In this section, we derive the critical condition for stability of the peer to peer networks after attack. In order to do that, we utilize the expression of the deformed degree distribution p'_k after removal of nodes. According to [28, 115], the critical condition for the stability of giant component can be expressed as

$$\kappa' = \frac{\langle k^2 \rangle'}{\langle k \rangle'} > 2\,, \tag{4.5}$$

where $\langle k\rangle'$ and $\langle k^2\rangle'$ refer to the first and second moments of the degree distribution after the attack. The critical condition $\kappa' = 2$ determines the point at which the network breaks down. To compute $\langle k\rangle'$ and $\langle k^2\rangle'$ of the modified network, we utilize the generating function $G_0(x) = \sum_k p'_k\, x^k$, which reads:

$$G_0(x) = \sum_{k=0}^{\infty}\sum_{q=k}^{\infty}\begin{pmatrix} q \\ k \end{pmatrix}\phi^{q-k}(1-\phi)^k p^s_q\, x^k\,. \tag{4.6}$$

After exchanging the order of the sum, the Binomial theorem can be applied, and we obtain:

$$G_0(x) = \sum_{k=0}^{\infty} p^s_k\, \left((x-1)(1-\phi)+1\right)^k. \tag{4.7}$$

From Eq. (4.7), the first two moments can be easily computed as $\langle k\rangle' = dG_0(1)/dx$ and $\langle k^2\rangle' = d^2G_0(1)/dx^2 + dG_0(1)/dx$, and the critical condition given by Eq. (4.5) takes the form:

$$(1-\phi)\,\frac{\langle k^2\rangle - \sum_{q=0}^{\infty} f_q\, p_q\, q^2}{\langle k\rangle - \sum_{q=0}^{\infty} f_q\, p_q\, q} + \phi = 2\,, \tag{4.8}$$

where $\langle k\rangle$ and $\langle k^2\rangle$ refer to the first and second moments of the degree distribution before the attack. Replacing ϕ by Eq. (4.2) and assuming $N >> 1$, we obtain

$$\sum_{k=0}^{\infty} kp_k(k(1-f_k)-(1-f_k)-1) = 0 \tag{4.9}$$

which is the critical condition of stability in any large scale uncorrelated peer to peer networks. Comparing Eqs. 4.9 and 3.15, we conclude that, this critical condition is exactly same as that developed in Chapter 3.

4.2 Effect of attacks upon the superpeer networks

In this section, we formally analyze the effect of attacks on the superpeer networks with the help of the developed framework. Two kinds of attacks, namely deterministic attack and degree dependent attack are discussed separately. The attack models are already described in Chapter 3. First of all, we show the effect of these attacks on the topological deformation of the network. This phenomenon has been modeled

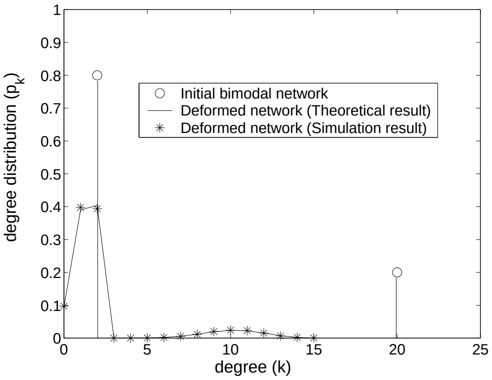


Figure 4.2: Topological deformation of the superpeer networks in face of deterministic attack. After the attack, 10% of nodes are removed. This 10% of nodes correspond to the 50% of the superpeer nodes whose degree is 20. The initial bimodal network and deformed network after attack are shown in the figure. The theoretically calculated degree distribution (p'_k) is verified through simulation.

using Eq. (4.4) and validated through simulations. Next we evaluate the stability of superpeer networks against these kinds of attacks and establish a relationship between them.

4.2.1 Analysis of deterministic attack

We consider superpeer networks with peer degree $k_l = 2$ and superpeer degree $k_m = 20$ and assume that 80% of nodes in the network are peers. Suppose 10% of nodes are removed through deterministic attack which signifies that 50% of superpeers get removed. We calculate the new degree distribution after attack (p'_k) by Eq. (4.4) and compare the results with simulation. Fig. 4.2 shows the good agreement between the theoretical and simulation results which confirms the success of our model.

Stability of the superpeer networks is challenged by attack on prominent peers or superpeers. In this section, we analyze the effect of this kind of targeted attack upon superpeer networks where r is the fraction of peers and rest are superpeers. In the case of targeted attack two cases may arise:

Case 1 Removal of a fraction of superpeers is sufficient to disintegrate the network.

Case 2 Removal of all the superpeers is not sufficient to disintegrate the network. Therefore, we need to remove some of the peer nodes along with the superpeers.

We analyze these two cases separately with the help of our analytical framework. First we consider the bimodal networks as our superpeer networks model. Next we extend the analysis for the more sophisticated mixed poisson networks.

Bimodal Networks

From Eq. (4.9) the critical condition for the stability of the superpeer networks can be rewritten as

$$\sum_{k=k_l,k_m} k(k-1)p_k q_k = \langle k \rangle \tag{4.10}$$

The equation can be further expanded as below to differentiate between peers and superpeers

$$k_l(k_l-1)p_{k_l}q_{k_l} + k_m(k_m-1)p_{k_m}q_{k_m} = \langle k \rangle \tag{4.11}$$

Case 1: In this case, removal of a fraction of superpeers is sufficient to disintegrate the network. If f_{sp} be the critical fraction of superpeer nodes, removal of which disintegrates the giant component, then $q_k = 1$ for $k = k_l$ and $q_k = 1 - f_{sp}$ for $k = k_m$. Hence according to Eq. (4.11),

$$\sum_{k=k_l} k(k-1)p_k + \sum_{k=k_m} k(k-1)p_k(1-f_{sp}) = \langle k \rangle$$

$$\Rightarrow f_{sp} = 1 - \frac{\langle k \rangle - k_l(k_l-1)p_{k_l}}{k_m(k_m-1)p_{k_m}}$$

As the fraction of superpeer nodes in the network is $(1-r)$, then percolation threshold for case 1 becomes $f_{tar} = (1-r) \times f_{sp}$

$$\Rightarrow f_{tar} = (1-r)\left(1 - \frac{\langle k \rangle - k_l(k_l-1)r}{k_m(k_m-1)(1-r)}\right) \tag{4.12}$$

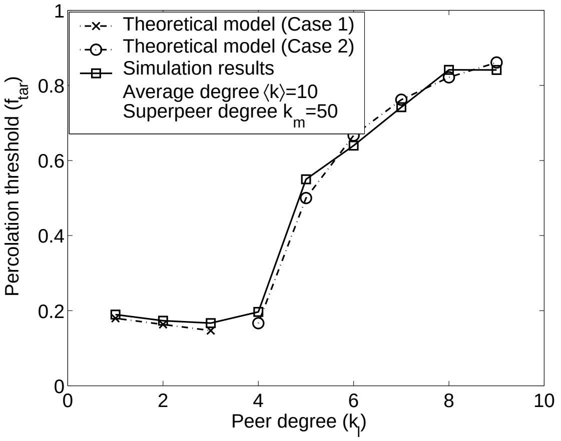


Figure 4.3: Stability of the superpeer networks in face of deterministic attack (Comparative study between theoretical and simulation results). Here X-axis represents the peer degree (k_l) and Y-axis represents the corresponding percolation threshold (f_{tar}). We keep the average degree $\langle k \rangle = 10$ and mean superpeer degree $\langle k_{sp} \rangle = 50$ fixed. Case 1 and case 2 of the theoretical model represent Eqs. (4.12) and (4.15) respectively.

Case 2: Here we have to remove f_p fraction of peer nodes along with all the superpeers to breakdown the network. Therefore $q_k = 1 - f_p$ for $k = k_l$ and $q_k = 0$ for $k = k_m$. Hence according to Eq. (4.11),

$$k_l(k_l - 1)p_{k_l}(1 - f_p) = \langle k \rangle \tag{4.13}$$

$$\Rightarrow f_p = 1 - \frac{\langle k \rangle}{k_l(k_l - 1)p_{k_l}} \tag{4.14}$$

Therefore the total fraction of nodes required to be removed to disintegrate the network for case 2 becomes $f_{tar} = rf_p + (1 - r)$.

$$\Rightarrow f_{tar} = r\left(1 - \frac{\langle k \rangle}{k_l(k_l - 1)r}\right) + (1 - r) \tag{4.15}$$

Transition point: The transition from case 1 to case 2 can be easily marked by observing the value of percolation threshold f_{tar}. While calculating using Eq. (4.12) (case 1), if the value of f_{tar} exceeds the fraction of superpeers in the network $(1-r)$, it indicates that removal of all the superpeers is not sufficient to disrupt the network. Hence subsequently we enter into case 2 and start using Eq. (4.15) to find percolation threshold.

We validate our theoretical model of attack on superpeer network with the help of simulation. During simulation, initially only high degree superpeer nodes in the network are removed gradually until the percolation point is reached. If the percolation point is not reached even after removing of all the superpeers, we remove a fraction of peers along with the superpeers to breakdown the network. We perform each experiment for 500 times and take the average of the percolation threshold obtained in each of them. Superpeer networks with average degree $\langle k\rangle = 10$ and superpeer degree $k_m = 50$ are considered for case study. We increase the peer degree k_l gradually (the peer fraction changes accordingly) and observe the change in the percolation threshold f_{tar} (Fig. 4.3).

Observations:

a. In the networks with peer degree $k_l = 1$, 2 and 3, the removal of only a fraction of superpeers causes breakdown thus making these networks more vulnerable. In fact, increase of peer degree from 1 to 2 and 3 further reduces the fraction of superpeers in the network. Subsequently, removal of only a small fraction of superpeer nodes causes breakdown of the network, hence makes networks with $k_l = 2$, 3 more vulnerable. In general, the vulnerability of a network against attack increases with the network heterogeneity. Since the increase in peer degree reduces the network heterogeneity, it would be expected that the attack vulnerability of a network will reduce with the increase in peer degree. But the opposite happens here. The slope of the Eq. (4.12) with respect to k_l becomes

$$\frac{\triangle f_{tar}}{\triangle k_l} = \frac{1}{M_2}\frac{(M_1 - k_l M_3 + M_4 k_l^2) - (M_5 - k_l)(2M_5 k_l - M_3)}{(M_5 - k_l)^2} \tag{4.16}$$

where M_1, M_2, M_3, M_4, M_5 are constants dependent on superpeer degree k_m and average degree $\langle k\rangle$. The slope of the curve at the points $k_l = 1, 2$ and 3 becomes negative which signifies that the attack vulnerability of the network increases with k_l. Along with the theoretical justification, this can also be explained by looking into the micro dynamics. In this zone (at $k_l = 2, 3$), although peers have a larger share in the

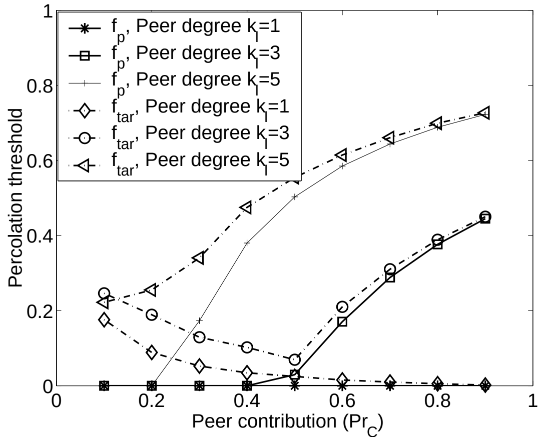


Figure 4.4: The plot represents the impact of peer contribution Pr_C upon the stability of the network against attack. f_p represents the fraction of peers required to be attacked to dissolve the network and f_{tar} indicates the corresponding percolation threshold.

network, yet it is not large enough to form effective connections within themselves. Therefore the stability of the network is still entirely dependent on the high degree superpeers, hence now attacking even a smaller fraction breaks down the network.

b. However as peer degree increases beyond 4, the transition from case 1 to case 2 occurs. In this region a fraction of peers is required to be removed even after removal of all the superpeers to dissolve the network. The slope of the Eq. (4.15) with respect to k_l becomes

$$\frac{\triangle f_{tar}}{\triangle k_l} = \frac{k}{k_l^2(k_l - 1)} + \frac{k}{k_l(k_l - 1)} \tag{4.17}$$

Hence the slope of the Eq. (4.15) becomes positive for any peer degree $k_l > 1$ which indicates that stability of the network increases with the increase of peer degree. In practice, the high degree peers connect among themselves and they are not entirely dependent on superpeers for connectivity. This results in the steep increase of stability of the network with peer degree $k_l \geq 5$.

Impact of peer contribution

Similar to churn, we investigate the impact of (pure) peer contribution upon stability of the network due to attack. In order to understand the influence of the degree of pure peers, we consider the networks with $k_l = 1, 3, 5$. Three sets of networks are generated having $k_l = 1, 3$ and 5, respectively, for individual peer contribution Pr_C $(0.1 \leq$

$Pr_C \leq 0.9$). In order to do that, we choose fraction of peers r uniformly at random and adjust superpeer degree k_m accordingly to keep the peer contribution Pr_C and peer degree k_l constant. This procedure is followed to generate one hundred networks for each set. We restrict superpeer degree $k_m \geq 20$ in order to generate realistic superpeer networks. We theoretically compute the percolation threshold (f_{tar}) and fraction of peers and superpeers required to be removed (f_p and f_{sp} respectively) for individual network and calculate their average for individual k_l. This expected fraction of peers required to be removed f_p and percolation threshold f_{tar} is plotted with respect to the peer contribution Pr_C (Fig. 4.4). The theoretical model is sufficient for analysis as the model has been already validated through simulation.

Observations:

1. It can be observed from Fig. 4.4 that superpeer networks having peer degree $k_l = 1$ can be disintegrated without attacking peers at all for any peer contribution Pr_C. This kind of attack belongs to case 1 of the attack model.

2. The peers of the superpeer networks having peer contribution $Pr_C \leq 0.2$ does not have any impact upon the stability of the network. This is true for low as well as high degree peers.

3. The influence of high degree peers increases with the increase of peer contribution. At $Pr_C = 0.3$, a fraction of peers is required to be removed to disintegrate the networks having peer degree $k_l = 5$. The impact of high degree peers upon the stability of the network becomes more eminent as peer contribution $Pr_C \geq 0.5$. In this region, a significant fraction of peers is required to be removed for all the networks having peer degree $k_l = 3, 5$. This kind of attack belongs to case 2 of the attack model.

4. Increase in peer contribution $Pr_C \geq 0.4$ brings the percolation threshold f_{tar} and fraction of peers needed to be attacked f_p close to each other which implies that stability of these networks is primarily dependent upon the stability of the peers.

5. It is interesting to observe that peer contribution Pr_C has two opposite effects upon stability of the networks depending on the peer degree k_l. The perco-

lation threshold f_{tar} increases with peer contribution Pr_C for $k_l = 3, 5$, but gradually reduces for $k_l = 1$. The reason behind this is, stability of the networks with peer degree $k_l = 1$ is entirely dependent upon superpeers. Since increase in peer contribution decreases superpeer contribution, it decreases stability of these networks also. On the other hand, peers having degree $k_l \geq 3$ have many connections among themselves, hence stability of these networks is more dependent upon peer contribution. Therefore, percolation threshold f_{tar} increases with peer contribution Pr_C.

6. Peer degree $k_l = 3$ exhibits some kind of trade off between the impact of peer and superpeer contribution upon stability. Superpeer contribution becomes more predominant for lower values of Pr_C ($Pr_C < 0.5$) which degrades the percolation threshold against attack. However as peer contribution Pr_C increases beyond 0.5, superpeer contribution reduces hence attacking peers along with superpeers is necessary to destroy the network. This increases the percolation threshold f_{tar} i.e. the stability of the network as well.

Mixed Poisson Networks

Similar to bimodal networks, in mixed poisson networks also we have two different cases. We analyze these two cases separately with the help of our analytical framework. From Eq. (4.9) the critical condition for the stability of the giant component can be rewritten as

$$\sum_{k=0}^{\infty} k(k-1)p_k q_k = \langle k \rangle$$

The equation can be further expanded as below to differentiate between peers and superpeers

$$\sum_{k=0}^{k_{max}-1} k(k-1)p_k q_k + \sum_{k=k_{max}}^{\infty} k(k-1)p_k q_k = \langle k \rangle \tag{4.18}$$

where all the nodes having degree less than k_{max} are peers and rest are superpeers.

Case 1: In this case, removal of a fraction of superpeers is sufficient to disintegrate the network. If f_{sp} be the critical fraction of superpeer nodes, removal of which disintegrates the giant component then $q_k = 1$ for $k < k_{max}$ and $q_k = 1 - f_{sp}$ for

$k \geq k_{max}$. Hence according to Eq. (4.18),

$$\sum_{k=0}^{k_{max}-1} k(k-1)p_k + \sum_{k=k_{max}}^{\infty} k(k-1)p_k(1-f_{sp}) = \langle k \rangle$$

$$\Rightarrow f_{sp} = 1 - \frac{\langle k \rangle - \sum_{k=0}^{k_{max}-1} k(k-1)p_k}{\sum_{k=k_{max}}^{\infty} k(k-1)p_k}$$

As the fraction of superpeer nodes in the network is $(1-r)$, then percolation threshold for case 1 becomes $f_t = (1-r) \times f_{sp}$

$$\begin{aligned} \Rightarrow f_t &= (1-r)\left(1 - \frac{\langle k \rangle - \sum_{k=0}^{k_{max}-1} k(k-1)p_k}{\sum_{k=k_{max}}^{\infty} k(k-1)p_k}\right) \\ &= (1-r)\left(1 - \frac{\langle k \rangle - r\sum_{k=0}^{\langle k_p \rangle+\delta} k(k-1)\frac{\langle k_p \rangle^k e^{-\langle k_p \rangle}}{k!}}{(1-r)\sum_{k=\langle k_p \rangle+\delta+1}^{\infty} k(k-1)\frac{\langle k_{sp} \rangle^k e^{-\langle k_{sp} \rangle}}{k!}}\right) \qquad (4.19) \end{aligned}$$

where mean peer degree $\langle k_p \rangle = \frac{\langle k \rangle - (1-r)\langle k_{sp} \rangle}{r}$ and we choose suitable value of δ depending on the standard deviation of the Poisson distribution. δ ensures the inclusion of all peer and superpeer degrees around their respective means $\langle k_p \rangle$ and $\langle k_{sp} \rangle$ during the calculation of above equations.

Case 2: Here we have to remove f_p fraction of peer nodes alongwith all the superpeers to breakdown the network. Therefore $q_k = 1 - f_p$ for $k < k_{max}$ and $q_k = 0$ for $k \geq k_{max}$. Hence according to Eq. (4.18),

$$\sum_{k=0}^{k_{max}-1} k(k-1)p_k(1-f_p) = \langle k \rangle$$

$$\Rightarrow f_p = 1 - \frac{\langle k \rangle}{\sum_{k=0}^{k_{max}-1} k(k-1)p_k}$$

Therefore the total fraction of nodes required to be removed to disintegrate the network for case 2 becomes $f_t = rf_p + (1-r)$.

$$\begin{aligned} \Rightarrow f_t &= r\left(1 - \frac{\langle k \rangle}{\sum_{k=0}^{k_{max}-1} k(k-1)p_k}\right) + (1-r) \\ &= r\left(1 - \frac{\langle k \rangle}{r\sum_{k=0}^{\langle k_p \rangle+\delta} k(k-1)\frac{\langle k_p \rangle^k e^{-\langle k_p \rangle}}{k!}}\right) + (1-r) \qquad (4.20) \end{aligned}$$

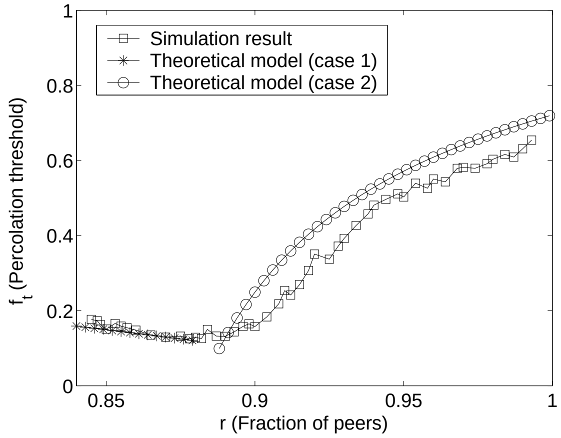


Figure 4.5: The above plot represents the behavior of the mixed poisson network in face of deterministic attack found experimentally and compares it with the proposed theoretical model. Here X-axis represents the fraction of peer nodes (r) that exist in the network and Y-axis represents the corresponding percolation threshold (f_t). We keep the average degree $\langle k \rangle = 5$ and mean superpeer degree $\langle k_{sp} \rangle = 30$ fixed. Case 1 and case 2 of the theoretical model represent Eqs. (4.19) and (4.20) respectively.

where mean peer degree $\langle k_p \rangle = \frac{\langle k \rangle - (1-r)\langle k_{sp} \rangle}{r}$.

Transition point: The transition from case 1 to case 2 can be easily marked by observing the value of percolation threshold f_t. While calculating using Eq. (4.19) (case 1), if the percolation threshold f_t exceeds the fraction of superpeers in the network $(1 - r)$, it indicates that removal of all the superpeers is not sufficient to disrupt the network. Hence subsequently we enter into case 2 and start using Eq. (4.20) to find percolation threshold.

We validate our theoretical model of attack on mixed poisson network with the help of simulation. In simulation, we consider a mixed poisson network with average degree $\langle k \rangle = 5$ and mean superpeer degree $\langle k_{sp} \rangle = 30$. We increase the fraction of peers gradually keeping average degree $\langle k \rangle = 5$ fixed and observe the change in the percolation threshold f_t (Fig. 4.5). It is important to note that when the fraction of superpeers in the network is high, it is possible to breakdown the network only by removing a fraction of superpeers and modeled as case 1 (Eq. (4.19)). But when the fraction of superpeers is below some threshold, a fraction of peers should be attacked alongwith the superpeers to stop percolation in the network and modeled as case 2

(Eq. (4.20)).

Summarization: In this section, the impact of deterministic attack on the stability of superpeer networks has been analyzed. We have shown that the networks having peer degree $k_l \leq 3$ are very much vulnerable and removal of only a small fraction of superpeers causes the breakdown of the network. But as the peer degree increases, the stability of the network increases as well. We have observed that peer contribution plays a major role in the network stability, specially for the networks with high peer degree (say $k_l \geq 3$). In this case, a fraction of peers are required to be removed along with all the superpeers in the network. However, depending upon the peer degree k_l, peer and superpeer contributions exhibit two opposite forces in percolation threshold due to their individual influence on the connectivity of the network. This phenomenon becomes much more predominant for the networks with $k_l \geq 3$.

Mixed poisson network is modeled as the superposition of two E-R graphs (with Poisson degree distributions) with two different average degrees. The major fraction of nodes in an E-R graph has degree close to the mean degree. Hence an E-R graph following Poisson degree distribution with mean degree $\langle k \rangle$ can be approximated by a regular graph with degree k. In order to simplify our calculation, we extend this approximation for the mixed poisson network. In this approximation, we model the superpeer networks using bimodal degree distribution instead of mixed poisson. Rigorous simulation results show that both of these networks namely bimodal networks and mixed poisson networks exhibit similar qualitative behavior under various node disturbances like failure and attack. We henceforth use bimodal network as the representative superpeer network for the analysis of degree dependent attack; since it is simple enough to understand, at the same time it captures the essential features of superpeer networks.

4.2.2 Analysis of degree dependent attack

In this kind of attack, the probability of removal of a node of degree k is directly proportional to k^γ where $\gamma \geq 0$ is a real number and represents the information available to the attacker about the topological structure of the network. Similar to the deterministic attack, in this case also we compute the deformed degree distribution p'_k after attack and validate the results through simulations. Without the loss of generality, we use bimodal network as the representative topology to model superpeer networks. We consider a superpeer network with peer degree $k_l = 2$ and superpeer degree $k_m = 10$ where 80% of the nodes are peers. The probability of removal of a node is proportional to its degree, i.e. $f_k = \frac{k}{k_m+1}$ (so $\gamma = 1$). The theoretically computed p'_k (using Eq. (4.4)) and simulation results are shown in Fig. 4.6. Next

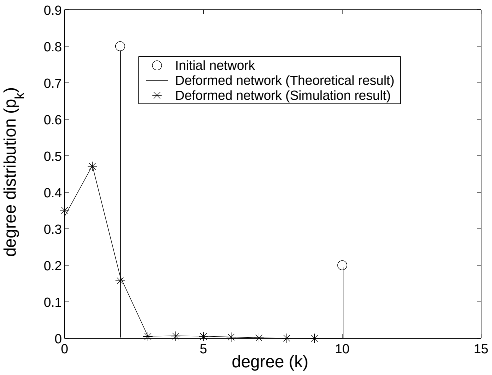


Figure 4.6: Topological deformation of the superpeer networks in face of degree dependent attack. The nodes are removed from the network with $f_k = \frac{k}{k_m+1}$. The initial bimodal network and the deformed network after attack p'_k are shown in the figure.

we analyze the effect of degree dependent attack upon the stability of the superpeer networks. With proper normalization, probability of removal of a node having degree k becomes $f_k = \frac{k^\gamma}{C}$ where C is the normalization constant.

As mentioned in bimodal degree distribution, let r be the fraction of peers with degree k_l while rest are superpeers of degree k_m. If $\langle k \rangle$ is the average degree of the network, then

$$p_{k_l} = r = \frac{k_m - \langle k \rangle}{k_m - k_l} \qquad p_{k_m} = (1 - r) = \frac{\langle k \rangle - k_l}{k_m - k_l}$$

From Eq. (4.9) the critical condition for the stability of the giant component can be rewritten as

$$\sum_{k=k_l,k_m} k(k-1)p_k(1 - f_k) = \langle k \rangle$$

$$\begin{aligned}\Rightarrow \quad & \langle k^{\gamma+2}\rangle - \langle k^{\gamma+1}\rangle = C(\langle k^2\rangle - 2\langle k\rangle) \\ \Rightarrow \quad & r k_l^{\gamma+1}(k_l - 1) + (1-r)k_m^{\gamma+1}(k_m - 1) = \\ & C(\langle k\rangle(k_m + k_l) - k_m - 2\langle k\rangle) \end{aligned} \tag{4.21}$$

where θ^{th} moment of the bimodal degree distribution can be written as $\langle k^\theta\rangle = k_m^\theta p_{k_m} + k_l^\theta p_{k_l}$. The solution of Eq. (4.21) yields a particular value of γ, say γ_c (termed as critical exponent) and the percolation threshold becomes

$$f_c^{\gamma_c} = r\frac{k_l^{\gamma_c}}{C} + (1-r)\frac{k_m^{\gamma_c}}{C} \tag{4.22}$$

In order to evaluate the disintegration point, proper assignment of the value of normalizing constant C is necessary. Since f_k should be ≤ 1 $\forall k$, hence the minimum value of $C = k_m^\gamma$. Assuming this condition, Eq. (4.21) becomes

$$\begin{aligned} r k_l^{\gamma+1}(k_l - 1) + (1-r)k_m^{\gamma+1}(k_m - 1) \quad & \geq \\ k_m^\gamma(\langle k\rangle(k_m + k_l) - k_m - 2\langle k\rangle) & \end{aligned} \tag{4.23}$$

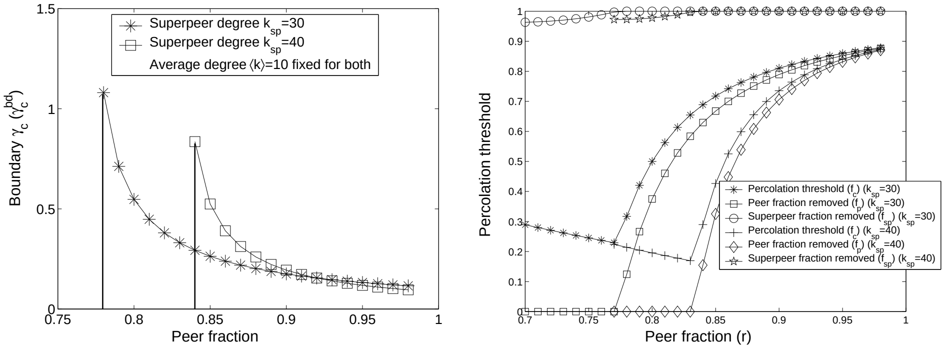
The solution set of the above inequality (say S_{γ_c}) can be bounded (where $0 \leq \gamma_c \leq \gamma_c^{bd}$) or unbounded (where $0 \leq \gamma_c \leq +\infty$). Each critical exponent $\gamma_c \in S_{\gamma_c}$ specifies the fraction of peers and superpeers required to be removed to breakdown the network. Assuming equality of Eq. (4.23) and hence obtaining minimum value of C, each γ_c results in the corresponding normalizing constant

$$C_{\gamma_c} = \frac{r k_l^{\gamma_c+1}(k_l - 1) + (1-r)k_m^{\gamma_c+1}(k_m - 1)}{\langle k\rangle(k_m + k_l) - k_m - 2\langle k\rangle} \tag{4.24}$$

Hence the fraction of peers and superpeers need to be attacked are

$$f_p^{\gamma_c} = \frac{k_l^{\gamma_c}}{C_{\gamma_c}} \qquad f_{sp}^{\gamma_c} = \frac{k_m^{\gamma_c}}{C_{\gamma_c}} \tag{4.25}$$

respectively and the total fraction of removed nodes $f_c^{\gamma_c}$ is obtained from Eq. (4.22). The $f_c^{\gamma_c}$ depends upon the critical exponent $\gamma_c \in S_{\gamma_c}$ and normalizing constant C_{γ_c}. The nature of the solution set S_{γ_c} has profound impact upon the behavior of $f_p^{\gamma_c}$, $f_{sp}^{\gamma_c}$ as well as $f_c^{\gamma_c}$. The breakdown of the network can be due to one of the three situations noted below.



(a) Behavior of γ_c^{bd} with respect to the change in peer fraction (r).

(b) Fraction of peers and superpeers required to be removed to breakdown the network and its impact upon percolation threshold f_c.

Figure 4.7: Case 1 of the degree dependent attack. The superpeer degree k_m is adjusted with the change of peer fraction r to keep the average degree fixed.

1. The removal of all the superpeers along with a fraction of peers.

2. The removal of only a fraction of superpeers.

3. The removal of some fraction of both superpeers and peers.

The above mentioned three cases are discussed one by one with example.

Case 1 : Removal of all superpeers along with a fraction of peers

Networks having bounded solution set S_{γ_c} where $0 \leq \gamma_c \leq \gamma_c^{bd}$ exhibit this kind of behavior at the maximum value of the solution $\gamma_c = \gamma_c^{bd}$. Here the fraction of superpeers removed become $f_{sp}^{\gamma_c^{bd}} = 1$ and fraction of peers removed $f_p^{\gamma_c^{bd}} = \frac{k_l^{\gamma_c^{bd}}}{C_{\gamma_c^{bd}}}$. We consider superpeer networks with superpeer degrees $k_m = 30, 40$ and average degree $\langle k \rangle = 10$ and theoretically study the stability of the networks due to the change in the peer fraction r. The results of the case study are noted in Fig. 4.7. It can be observed that the solution set of these networks upto a threshold peer fraction r_c,

($r_c = 0.78$ and 0.84 for $k_m = 30$ and $k_m = 40$ respectively) remains unbounded. The bounded solution set is observed for the networks with $r \geq r_c$ and the behavior of the boundary critical exponent γ_c^{bd} due to the change of peer fraction r is shown in Fig. 4.7(a). The fraction of peers and superpeers needed to be attacked for these networks is presented in Fig. 4.7(b). These networks exhibit the properties of case 1 of degree dependent attack, hence the removal of all the superpeers is necessary to disintegrate the network along with a fraction of peers. Fig. 4.7(b) also represents some instances of case 2 where only some fraction of superpeers are needed to be removed ($r < r_c$).

The main findings are listed below

a. Impact upon the fraction of peers removed

The increase in peer fraction slowly decreases γ_c^{bd} (Fig. 4.7(a)) which in turn gradually increases the fraction of peers removed $f_p^{\gamma_c^{bd}}$ (Fig. 4.7(b)). The amount of removal of peers also depends upon the superpeer degree k_m. The increase in the superpeer degree reduces the role of peers in determining the stability of the network. Hence fraction of peers required to be removed $f_p^{\gamma_c^{bd}}$ reduces with increase in k_m.

b. Impact upon percolation threshold

Let the percolation threshold for the networks having peer fraction r_1 and r_2 (where $r_1 < r_2$) be $f_{c1}^{\gamma_c^{bd}}$ and $f_{c2}^{\gamma_c^{bd}}$ respectively. Hence the percolation threshold for these two networks are

$$f_{c_1}^{\gamma_c^{bd}} = r_1 f_{p_1}^{\gamma_c^{bd}} + (1 - r_1) \tag{4.26}$$

$$f_{c_2}^{\gamma_c^{bd}} = r_2 f_{p_2}^{\gamma_c^{bd}} + (1 - r_2) \tag{4.27}$$

Therefore the change in the percolation threshold when the peer fraction changes from r_1 to r_2 is

$$\begin{aligned} f_{c_1}^{\gamma_c^{bd}} - f_{c_2}^{\gamma_c^{bd}} = \triangle f_c^{\gamma_c^{bd}} &= r_1 f_{p_1}^{\gamma_c^{bd}} - r_2 f_{p_2}^{\gamma_c^{bd}} - (r_1 - r_2) \\ &= \triangle\left(r f_p^{\gamma_c^{bd}}\right) - \triangle r \end{aligned} \tag{4.28}$$

The Eq. (4.28) shows that the change of percolation threshold $f_c^{\gamma_c^{bd}}$ is influenced by two opposite forces; on one hand the increase of peer fraction r (from r_1 to r_2) in the network makes $\triangle r < 0$ that increases $\triangle f_c^{\gamma_c^{bd}}$. On the other hand, this increase in r increases the fraction of peers required to be removed (Fig. 4.7(b)) which makes

$\triangle\left(rf_p^{\gamma_c^{bd}}\right) < 0$. Depending upon the weightage of influence, $\triangle f_c^{\gamma_c^{bd}}$ (and subsequently $f_c^{\gamma_c^{bd}}$) either decreases or increases. For $r < r_c$, the $rf_p^{\gamma_c^{bd}}$ remains 0, hence $f_c^{\gamma_c^{bd}}$ decreases with r. When peer fraction $r \geq r_c$, due to the finite value of $f_p^{\gamma_c^{bd}}$, the $f_c^{\gamma_c^{bd}}$ increases.

Case 2 : Removal of only a fraction of superpeers

Some networks have unbounded solution set S_{γ_c} where $0 \leq \gamma_c \leq +\infty$. As $\gamma_c \to \infty$, $f_p^{\gamma_c}$ converges to 0 and $f_{sp}^{\gamma_c}$ converges to some x where $0 < x < 1$. This illustrates the case 2 of degree dependent attack where removal of only a fraction of superpeers is sufficient to disintegrate the network. The case study is performed with a network having superpeer degree $k_m = 25$, average degree $\langle k \rangle = 5$ and peer degree $k_l = 2$. The results are validated with the help of simulation. We plot the theoretically calculated (Eqs. (4.24), (4.25)) fraction of peers and superpeers required to be removed to breakdown the network for each critical exponent γ_c (Fig. 4.8). In simulation, we initially remove the fraction of superpeers $f_{sp}^{\gamma_c}$ which has been predicted theoretically and then start removing peers gradually to breakdown the network. The minimum peer fraction, removal of which causes the breakdown of the network corresponds to the simulated $f_p^{\gamma_c}$. We perform the simulation on graphs of 5000 nodes and repeat each experiment for 500 times and take the average of the removed peer fraction. We compare simulated results with theoretically calculated $f_p^{\gamma_c}$ (Fig. 4.8). The interesting findings are noted below.

a. The fraction of peers removed $f_p^{\gamma_c}$ gradually decreases with the increase of the critical exponent γ_c, which in turn decreases the value of $f_c^{\gamma_c}$. As $\gamma_c \to \infty$, the $f_p^{\gamma_c} \to 0$ with $f_{sp}^{\gamma_c} \to x$ (where $0 < x < 1$) and $f_{sp}^{\gamma_c}$, $f_c^{\gamma_c}$ both converges to some steady value. This signifies that the removal of only a fraction of superpeers is sufficient to breakdown the network (Fig. 4.8).

b. In Fig. 4.7(a), the nonexistence of the boundary critical exponent γ_c^{bd} for the networks having peer fraction $r < r_c$ signifies that the solution set of these networks is unbounded and the percolation process belongs to case 2. It can be observed that the fraction of peers required to be removed for these networks becomes zero (Fig. 4.7(b)) and removal of only a fraction of superpeers disintegrates the network.

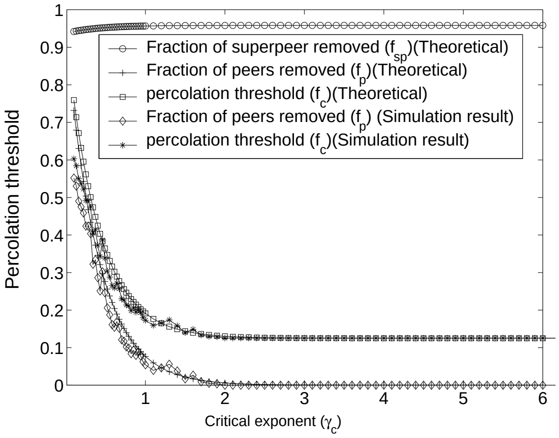


Figure 4.8: The above plot illustrates the case 2 of degree dependent attack.

c. It is important to note that removal of only a fraction of superpeers is sufficient to disintegrate any network with peer degree $k_l = 1$ and 2 irrespective of the superpeer degree and its fraction. Mathematically it can be explained as follows. For $k_l \leq 2$, $2k_l \geq k_l^2$

$$
\begin{aligned}
&\Rightarrow \quad 2rk_l \geq rk_l^2 \\
&\Rightarrow \quad (1-r)k_m + 2rk_l - rk_l^2 \geq 0 \\
&\Rightarrow \quad (1-r)k_m(k_m-1) \geq \langle k\rangle(k_m+k_l) - k_m - 2\langle k\rangle \\
&\Rightarrow \quad rk_l^{\gamma+1}(k_l-1) + (1-r)k_m^{\gamma+1}(k_m-1) \geq \\
&\qquad\;\; k_m^{\gamma}(\langle k\rangle(k_m+k_l) - k_m - 2\langle k\rangle)
\end{aligned}
$$

This is exactly the inequality that we get in Eq. 4.23. This inequality is essentially the condition for breakdown of the superpeer network. Since the above inequality holds for any values of γ, it indicates that any network with $k_l = 1, 2$ has unbounded solution set.

Case 3 : Removal of some fraction of both peers and superpeers

Degree dependent attack allows to disintegrate the network by removing a fraction of both peers and superpeers. Intermediate critical exponents ($\gamma_c \in S_{\gamma_c}$ and $\gamma_c \neq \gamma_c^{bd}$)

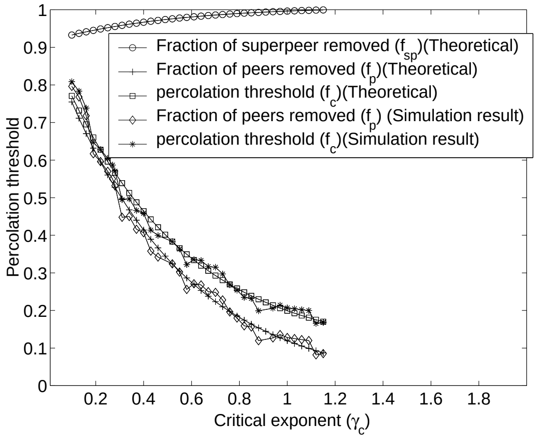


Figure 4.9: The above plot illustrates the case 3 of the degree dependent attack.

signify the fractional removal of both peers and superpeers. We calculate the amount of peers and superpeers needed to be removed to dissolve the network due to the change in γ_c. We deduce the results for a network having superpeer degree $k_m = 25$, average degree $\langle k \rangle = 5$ and peer degree $k_l = 3$. Results are also validated with the help of simulation (Fig. 4.9). The simulation set up is same as that described for case 2 of the degree dependent attack.

Observations:

a. Our analytical results show that this network has bounded solution set S_{γ_c} of the inequality (4.23) and all the critical exponents γ_c less than the boundary critical exponent $\gamma_c^{bd} = 1.171$ results in this kind of breakdown. It is evident from both theoretical and simulation results that the removal of any combination of $f_p^{\gamma_c}$, $f_{sp}^{\gamma_c}$ (obtained from the curves in Fig. 4.9) where $0 \leq \gamma_c < \gamma_c^{bd}$, results in the breakdown of the network.

b. Networks with unbounded solution set (Fig. 4.8) have finite values of γ_c ($\gamma_c < 2$) where the removal of both fraction of peers and superpeers are necessary to disintegrate the network.

Summarization: In this section, the impact of degree dependent attack on the stability of the superpeer networks has been discussed in details. We have formulated the critical condition for network stability and subsequently obtained the critical exponent γ_c. This critical exponent γ_c and the normalizing constant C_{γ_c} determine the amount of peers and superpeers required to be removed to breakdown the network. Interestingly, we also find that the removal of only a fraction of superpeers is suffi-

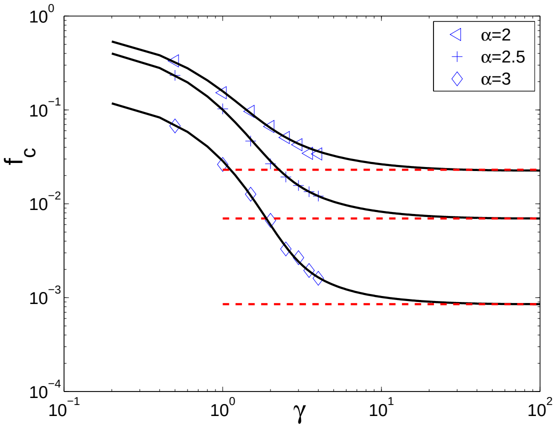


Figure 4.10: The above plot illustrates the change in percolation threshold f_c with the change of attack exponent γ. Three different scale free networks ($p_k \sim k^{-\alpha}$) with $\alpha = 2, 2.5$ and 3 have been considered. Curves represent the theoretical results whereas the symbols show the simulation results. The agreement between theoretical and simulation results (with $N = 10^5$) shows the success of Eq. (4.31). The dashed lines indicate the line of convergence of f_c calculated using Eq. (4.31) at $\gamma \to \infty$.

cient to disintegrate any network with peer degree $k_l = 1$ and 2 irrespective of the superpeer degree and its fraction [112].

One of the major contributions of this section is that, we have been able to provide a ***uniform attack framework*** (through degree dependent attack $f_k \sim k^\gamma$) which besides providing a flexibility in deciding attack strategy (through γ) also captures the essential features of deterministic attack. Case 1 and case 2 of the degree dependent attack resemble exactly the case 2 and case 1 of the deterministic attack respectively. In addition, $\gamma = 0$ and $\gamma < 0$ essentially model the degree independent and degree dependent failures respectively which have been illustrated in Chapter 3.

4.2.3 Physical interpretation of the attack exponent γ

The availability of the generalized attack model $f_k \sim k^\gamma$ immediately points to the importance of analyzing the attack parameter γ which signifies the information available to the attacker to breakdown the network [55]. As we know, the generalized attack can be represented as $f_k = \frac{k^\gamma}{C}$ where C is the normalizing constant. Clearly in the case of $\gamma > 0$, high degree nodes are removed with higher probability. Under this kind of generalized attack, the critical condition for stability of the large scale networks ($N \to \infty$) with degree distribution p_k can be expressed according to Eq. (4.9) as follows:

$$\langle k^2 \rangle - 2\langle k \rangle + \frac{[\langle k^{1+\gamma} \rangle - \langle k^{2+\gamma} \rangle]}{C} = 0\,, \tag{4.29}$$

where $\langle k^\omega \rangle$ is defined as $\langle k^\omega \rangle = \sum_k k^\omega\, p_k$. In consequence, the critical value of C that breaks down the network (termed as 'percolating C') simply reads:

$$C = \frac{\langle k^{2+\gamma} \rangle - \langle k^{1+\gamma} \rangle}{\langle k^2 \rangle - 2\langle k \rangle}. \tag{4.30}$$

The fraction of removed nodes f after an attack becomes $f = \sum_k p_k f_k$. Interestingly, for a given value of γ, the value of C obtained from Eq. (4.30) may not be feasible if $f_k = \frac{k^\gamma}{C} > 1$. This implies that an attack of the form $f_k = \frac{k^\gamma}{C}$ is unable to destroy the network. Given an attack characterized by an exponent γ, and using Eq. (4.30), the critical fraction of nodes that is required to remove in order to destroy the network is given by

$$f_c = \frac{\langle k^2 \rangle - 2\langle k \rangle}{\langle k^{2+\gamma} \rangle - \langle k^{1+\gamma} \rangle} \langle k^\gamma \rangle\,. \tag{4.31}$$

Eq. 4.31 is a generalized expression and can be applicable for any kind of network. However, the concept of topology information γ becomes more relevant for the network with continuous degree distribution, rather than the network consisting only two distinct degrees. Hence, next we perform a case study for the scale free networks where degree distribution follows $p_k \sim k^{-\alpha}$ with a maximum degree k_M. Fig 4.10 illustrates the behavior of the percolation threshold f_c of the scale free networks due to the change in the attack exponent γ. It also shows a comparison between Eq. (4.31) and stochastic simulations performed on the networks of size 10^5 with 500 realizations. In order to find the simulated value of percolating C as well as percolation threshold f_c, we have followed the method described in Chapter 3. As expected, random failure ($\gamma = 0$) requires high attack intensity that increases percolation threshold. However as $\gamma \to \infty$,

$$\begin{aligned} f_c &\to (\langle k^2 \rangle - 2\langle k \rangle) \lim_{\gamma \to \infty} \frac{\langle k^{\gamma} \rangle}{\langle k^{2+\gamma} \rangle - \langle k^{1+\gamma} \rangle} \\ \Rightarrow f_c &\to h(\alpha) \frac{1}{k_M (k_M - 1)} \end{aligned} \tag{4.32}$$

where $h(\alpha)(= \langle k^2 \rangle - 2\langle k \rangle)$ is a constant function of power law exponent α and maximum degree of the network k_M. Hence as information about the network (γ) increases, f_c decreases and converges to some constant value. The analysis of this attack has revealed that in scale free networks an increase of γ leads to a decrease of the critical fraction of nodes that must be removed to disintegrate the network; i.e. a decrease in the percolation threshold f_c. However, after a threshold γ, the percolation threshold f_c reaches to some constant value and does not decrease further.

4.2.4 Impact of network size on the percolation threshold

Till now, our work has focused on analyzing the stability of large scale networks; this is in line with the general trend. Hence, the percolation threshold f_c remains independent of the network size N. However, the framework developed in this chapter provides us the flexibility to understand the stability of small scale networks also. In this section, we illustrate the effect of network size N upon the percolation threshold $f_c(N)$. In section 4.1.1, we compute the probability ϕ of finding an edge in the surviving subset S that is connected to a node of other subset R (Fig. 4.1) as

$$\phi = \frac{E}{\sum_{i=0}^{\infty} i\, n_i\, (1 - f_i)} = \frac{\sum_{i=0}^{\infty} i\ p_i\, f_i}{(\sum_{k=0}^{\infty} k\, p_k) - 1/N}\,. \tag{4.33}$$

Following section 4.1.2, we find that the critical condition for the disintegration of the finite size networks can be expressed as

$$\left(\sum_k k p_k (1 - f_k)\right) \left(\sum_k p_k k^2 (1 - f_k) + \sum_k k p_k (f_k - 2)\right) + \frac{1}{N}\left(\sum_k k p_k (1 - f_k)(2 - k)\right) = 0 \tag{4.34}$$

Next we customize Eq. 4.34 for random failure by substituting $f_k = f$. Subsequently the percolation threshold for finite size network becomes

$$f_c(N) = \left(1 - \frac{1}{\frac{\langle k^2\rangle}{\langle k\rangle} - 1}\right) + \frac{1}{N}\left(\frac{2 - \langle k^2\rangle/\langle k\rangle}{\langle k^2\rangle - \langle k\rangle}\right) \tag{4.35}$$

As network size $N \to \infty$, the expression of percolation threshold for random failure reduces to

$$f_c^{\infty} = 1 - \frac{1}{\frac{\langle k^2\rangle}{\langle k\rangle} - 1} \tag{4.36}$$

which converges to Eq. (3.17) of Chapter 3.

Although Eq. (4.35) is a generalized expression, we show the results for Erdos-Renyi graph where the distinction between the finite and infinite size networks becomes nicely evident. We perform analysis on the E-R graph of finite size N with average degree $\langle k\rangle = 3$. Fig. 4.11 shows a comparative study between the percolation thresholds calculated from Eq. 4.35 (where we consider the network size N) and from Eq. 4.36 (where f_c is invariant of network size) and results obtained from stochastic simulation. As Eq. 4.36 does not take the network size under consideration, f_c^{∞} takes a constant value for a specific network configuration. However, $f_c(N)$ calculated from Eq. 4.35 takes a lower value for small sized networks and gradually increases with increase in N. The observed deviation between $f_c(N)$ and simulation results can be arguably attributed to clustering effects, which have been ignored in the current approach.

4.3 Effect of attacks upon the commercial Gnutella Networks

In the previous sections, we have modeled the superpeer networks as various theoretical random graphs and validated our theoretically derived results through stochastic simulation. In this section, we choose the commercially popular peer-to-peer network, Gnutella as a case study and examine its stability in face of attacks. In section 4.1.2, we have shown that the measurement of network stability primarily depends upon

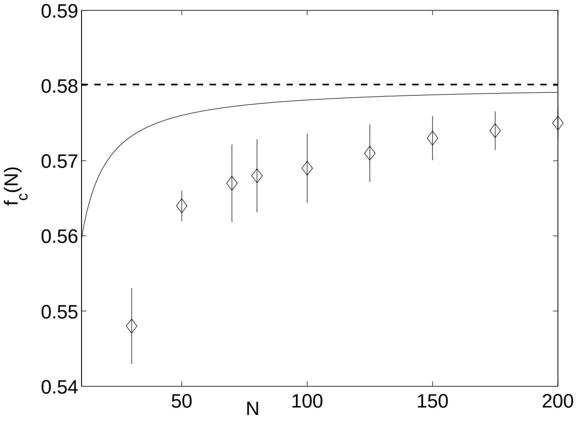
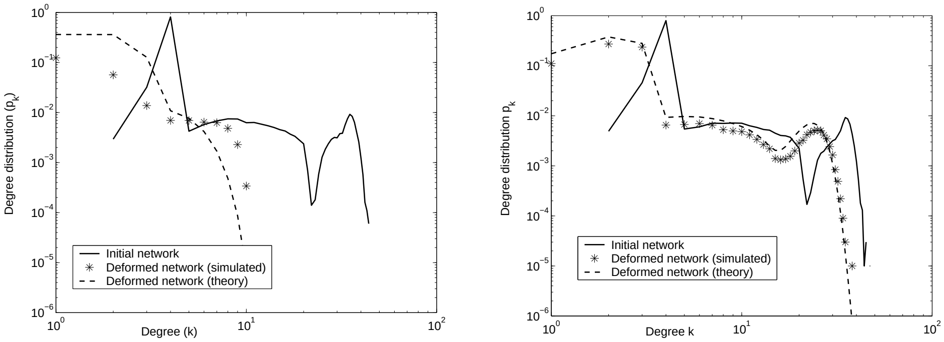


Figure 4.11: The figure illustrates the impact of network size N upon the percolation threshold f_c. The symbols represent the f_c obtained from stochastic simulation with a large number of realizations. The dashed line shows the percolation threshold calculated by Eq. (3.17) first proposed in [28] where f_c remains invariant with network size. The solid line shows the f_c calculated according to Eq. (4.35). The nature of the curve of Eq. (4.35) matches with the simulation however the results are not exact.

the deformed degree distribution p'_k after attack. Hence, in this section we focus on the accurate calculation of p'_k for Gnutella networks. We perform a comparative study of the p'_k obtained from the experiments on Gnutella networks with the results calculated from the analytical framework.

4.3.1 Attacks on Gnutella networks

In Chapter 3, we have described the generation of Gnutella networks following (a) bootstrapping protocol (b) topological snapshot. In this section, we refer the Gnutella network generated from bootstrapping protocol as 'Gnutella A' and Gnutella network generated from the topological snapshot as 'Gnutella B' and simulate deterministic attack and random failure on these two networks. We simulate the 'Gnutella A' network of $N = 5000$ nodes and all nodes in the network having degree more than 10 are removed in deterministic attack scenario. In random failure, 20% nodes in the network are removed randomly. The experiment is performed for 500 realizations

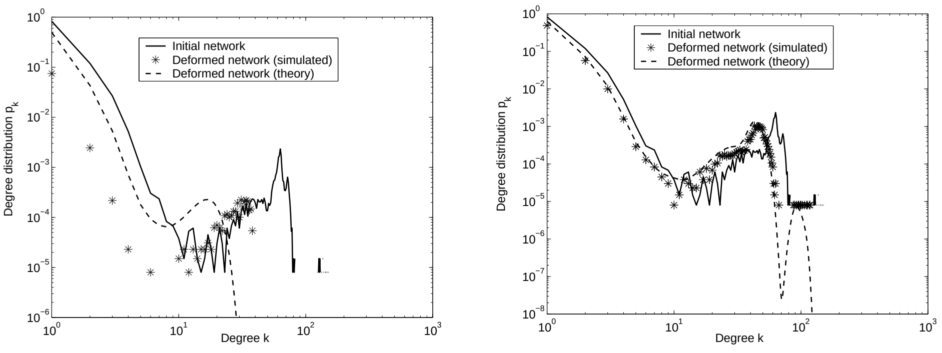


(a) The degree distribution of the deformed Gnutella network after deterministic attack. Here all the nodes in the network having degree greater that $k_{cut} = 10$ are removed.

(b) The degree distribution of the deformed Gnutella network after random failure. Here 20% of the nodes are randomly removed from the network.

Figure 4.12: The above plots show the topological impact of deterministic attack and random failure upon the simulated Gnutella A network of 5000 nodes. A comparative study of the simulation results with our theoretical model is performed.

and the average of the deformed degree distribution ($p'_{k_{sim}}$) and percolation threshold (f_{sim}) are calculated. We plot the degree distribution of the initial (p_k) and deformed network ($p'_{k_{sim}}$) in Fig. 4.12 and compare the simulation results with the theoretically calculated $p'_{k_{theory}}$ according to Eq. (4.4). Similarly, we mount a deterministic attack on 'Gnutella B' network where all the nodes in the network having degree more than 40 are removed. In random failure, 20% nodes in the network are removed randomly. The comparative study of the deformed degree distribution $p'_{k_{sim}}$ obtained from simulation with the theoretical model (Eq. (4.4)) has been done for these two kinds of node disturbances (Fig. 4.13). We observe that in both topologies (Gnutella A and B), the proposed theoretical model provides a reasonable approximation of the topological changes in the network under random failure (Fig. 4.12(b), Fig. 4.13(b)) however there is a deviation in case of deterministic attack (Fig. 4.12(a), Fig. 4.13(a)). We quantify the deviation of the theoretically predicted result from simulation in two different perspectives. First, we calculate the deviation in the individual $p_k \forall k$ (micro level deviation), second, the deviation in the average degree (macro level deviation). In order



(a) The degree distribution of the deformed Gnutella network after deterministic attack. Here all the nodes in the network having degree greater that $k_{cut} = 40$ are removed.

(b) The degree distribution of the deformed Gnutella network after random failure. Here 20% of the nodes are removed from the network randomly.

Figure 4.13: The above plots show the effect of attack and failure upon the Gnutella B network simulated from the topological snapshot taken during September 2004. The network is of the size of $1,31,869$ nodes. A comparative study of the simulation results with our theoretical model is performed.

to quantify the deviation of individual $p_k, \forall k$ for Gnutella A network, we calculate the deviation parameter dev_A in the following manner. We compute $p'_{k_{sim}}$ and $p'_{k_{theory}}$ for individual degree k and subsequently derive their difference $diff_k = |p'_{k_{sim}} - p'_{k_{theory}}|$. The overall deviation (dev_A) is calculated from $\frac{\sum_k diff_k}{max(k)}$. Similarly we calculate the deviation parameter dev_B for the Gnutella B network. We find that the deviation parameter $dev_A = 0.0284$ in the Gnutella A network is higher than the Gnutella B network, $dev_B = 0.0219$. Next we show the deviation in the theoretically and experimentally calculated average degree of the Gnutella network after deterministic attack. In Gnutella A and B networks, the average degree of the initial network is 5.6191 and 2.4359 respectively. After attack, the new average degree obtained from simulation becomes $Avg\_deg^A_{sim} = 0.4858$ and $Avg\_deg^B_{sim} = 0.1608$ respectively for Gnutella A and B network. However the theoretically calculated average degree for these two networks show higher values than simulation ($Avg\_deg^A_{theory} = 1.5917$ and

$Avg\_deg^B_{theory} = 0.6617$). We believe that the observed deviation between theoretical and simulation results are due to the presence of degree-degree correlation in the network which was not present in the random graphs. We first formally define the degree-degree correlation and then examine its precise role.

Defining degree-degree correlation

Degree-degree correlation is defined as the probability of attachment of a source node to the target node given the present degree of the source/target node. Many networks show "assortative mixing" on their degrees, i.e., a preference for high-degree nodes to attach to other high-degree nodes in the network. Others show "dis-assortative mixing" where high degree nodes attach to low degree ones. In [123], this property has been conveniently measured by means of a single normalized index, the assortativity coefficient[2]. In our simulation, the Gnutella networks generated through the bootstrapping protocol (Gnutella A) as well as topological snapshot (Gnutella B) exhibit dis-assortativity (negative assortativity). The average assortativity of the Gnutella A for 500 realizations becomes $\alpha = -0.6749$ whereas the Gnutella B has $\alpha = -0.6318$. The deviation of the theoretical results from simulation for Gnutella A ($dev_A = 0.0284$) is more than the Gnutella B network ($dev_B = 0.0219$) as well as Gnutella A has lower assortativity than Gnutella B. This indicates some sort of relationship between the deviation and assortativity. The precise role of assortativity is investigated next.

Role of assortativity

In this section, we intuitively explain the deviation between the theoretical and simulation results in assortative network. First we explain the impact of assortativity on

[2] Degree-degree correlation of a network is formally defined through assortativity coefficient α [123] such that

$$\alpha = \frac{M^{-1}\sum_i j_i k_i - [M^{-1}\sum \frac{1}{2}(j_i + k_i)]^2}{M^{-1}\sum_i \frac{1}{2}(j_i^2 + k_i^2) - [M^{-1}\sum \frac{1}{2}(j_i + k_i)]^2}$$

where j_i, k_i are the degrees of the vertices at the ends of the i^{th} edge, with $i = 1...M$ (M is the total number of edges in the network).

the average degree of the network.

Impact of assortativity on the average degree

A given attack on some assortative network changes the average degree (density) of the network, and the amount of change depends upon the assortative nature of the network. In Fig. 4.1, we find that two types of edges originate from the nodes of the removed set R; (a) one set of edges whose other end is also connected to the nodes of set R (say E_R) (b) another set of edges whose other end is connected to the nodes of set S (say E). For any given attack f_k^{atk}, the number of nodes in set R will be same for all networks. Let us assume that due to attack f_k^{atk} on a given network, the number of tips removed only from the nodes of removed set R is $\widehat{R_{tips}}$ and E is the number of tips removed from the set S. The number of edge tips removed will be the summation of $\widehat{R_{tips}}$ and E. Hence, the total number of edges removed from the network after attack becomes $\frac{E+\widehat{R_{tips}}}{2}$. $\widehat{R_{tips}}$ will be a constant across all networks (it is directly dependent on the number of nodes removed); therefore the number of edges removed will be directly dependent upon the value of E. Subsequently, the number of edges survived in the network after the attack f_k^{atk} may be expressed as

$$E_{new} = E_{tot} - \frac{E + \widehat{R_{tips}}}{2} \tag{4.37}$$

The value of E (number of edges running between the set S and R) depends on the assortativity of the network. In case of deterministic attack in assortative network, most of the high degree nodes (in R) are connected among themselves (making E_R quite high), hence a very small number of edges E are connected to the set S. Using Eq. 4.37, we find that the removal of few E edges keeps the network quite dense with high average degree. However, in disassortative network, most of the edges E run between S (low degree nodes) and R (high degree nodes) and there exits few links E_R connecting the high degree nodes of set R. Subsequently, the removal of large number of E edges reduces the average degree. Hence $E_{new}(assort) > E_{new}(uncorr) > E_{new}(disassort)$.

Intuitive justification behind $Avg\_deg_{theory} > Avg\_deg_{sim}$ against attack

We simulate an attack on Gnutella networks (a disassortative network) such that most of the high degree nodes are removed. As explained, removal of high degree nodes removes the large number of edges running between set S and R , say E_{sim} (E obtained from simulation). On the other hand, in theoretically calculated E (according to the Eq. (4.1)), say E_{theory}, we assume that the network is uncorrelated in nature, hence there is an equal/uniform probability that the other end of the removed tip (in set R) is connected to the nodes in the set S and set R. Hence the total number of edges running between the set S and set R, calculated theoretically (E_{theory}) is less than E_{sim}. This difference in the estimation of E (E_{theory} and E_{sim}) affects the number of survived edges E_{new} (Eq. 4.37) in the survived network. More specifically, in the theoretical calculation, the amount of reduction of the average degree of the survived network after attack is underestimated than that of the simulation. Hence after the given attack, the simulated network ($p'_{k_{sim}}$) becomes more sparse than the theoretically calculated network ($p'_{k_{theory}}$). Subsequently, $Avg\_deg_{theory} > Avg\_deg_{sim}$. This directly answers the question why for Gnutella network, $Avg\_deg_{theory} > Avg\_deg_{sim}$ where *theory* signifies the uncorrelated network and *sim* signifies disassortative network.

Assortativity does not have any impact on random failure

However it is interesting to observe in Fig. 4.12(b) and Fig. 4.13(b) that although assortativity takes a major role in attack, it does not have any influence in random failure. In random failure, the nodes in the set S and R are placed independent of their degree, hence high and low degree nodes are uniformly distributed in those sets. Subsequently, there is an equal/uniform probability that the other end of the edge connected to a node of the removed set R is linked with either a node of set S or of set R. In this way, the effect of assortativity becomes nullified in face of random failure.

In the next section, we utilize this intuitive understanding to refine and rectify our analytical framework so that it becomes applicable to the correlated networks also.

4.4 Stability analysis for degree correlated networks

In the previous section, we find that our theoretical framework is not able to explain the exact behavior of Gnutella network in face of deterministic attack. However, we have presented an intuitive explanation for the deviation of the theoretically computed results from the simulation. In this section we refine our framework, developed in section 4.1 to include correlated networks and examine its applicability on Gnutella network.

4.4.1 Deformed topology after attack

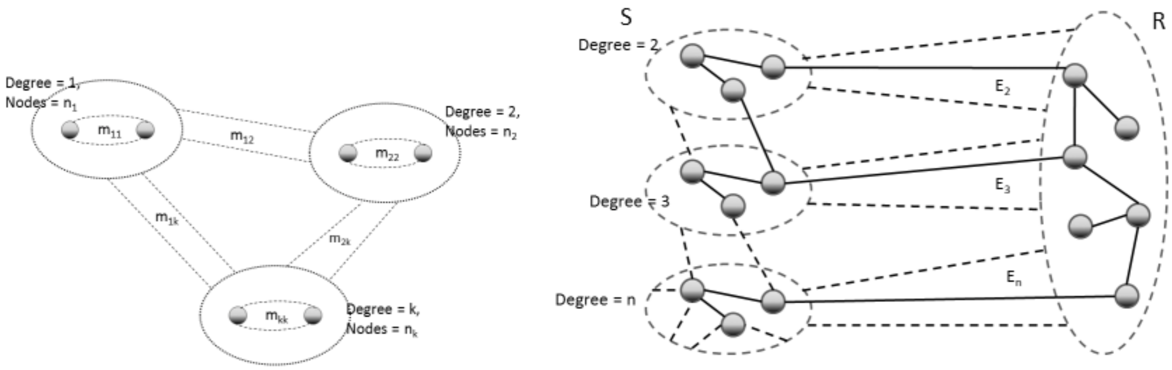
In this section, we modify the expression (derived in section 4.1.1) of deformed degree distribution p'_k to make it suitable for degree correlated networks. The degree-degree correlation information of a network with maximum degree k_M is represented by the correlation matrix M as follows

$$M = \begin{pmatrix} m_{11} & m_{12} & m_{13} & ... & m_{1k_M} \\ m_{21} & m_{22} & m_{23} & ... & m_{2k_M} \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ m_{k_M1} & m_{k_M2} & m_{k_M3} & ... & m_{k_Mk_M} \end{pmatrix}$$

In this correlation matrix M, each element m_{jk} represents the fraction of total edges that exist between nodes of degree j and nodes of degree k (Fig. 4.14(a)). We frame the attack on the network in the same manner as explained in the section 4.1.1. The attack on the network divides the graph into two sets of nodes: one set containing the surviving nodes S and another set containing the nodes to be removed R as shown in the Fig. (4.14(b)).

E_j instead of E

In section 4.1.1, we have calculated E which represents the number of edges running between set S and R. It is also the number of tips that is going to be removed from the nodes of the set S. The expression of E in Eq.(4.1) gives correct approximation for an uncorrelated network as the edge connectivity between a node of set R and any node of set S is equally probable. But in case of a degree correlated network, the probability of an edge between a node of degree i and a node of degree j is given by m_{ij} element of the correlation matrix M. Hence instead of calculating E we calculate E_j which indicates the number of edges connected between nodes of degree j in the set S and the nodes of any degree in the set R (Fig. 4.14(b)). Hence the total number of edges connected between the set S and R, that are going to be removed is given by $E = \sum_{j=0}^{k_M} E_j$. The expression for E_j can be formulated in the following way.



(a) The degree correlation in the network represented by the elements of the assortativity matrix M

(b) The dissection of a correlated network into two sets S and R due to the attack on the network.

Figure 4.14: Degree correlation present in the network and its implication on attack.

The total number of edge tips connected to the k degree nodes in set R can be expressed as kn_kf_k. Therefore, the number of edge tips connected to the j degree nodes of the network whose other end is connected to the k degree node of set R becomes $m'_{jk}kn_kf_k$. The fraction m'_{jk} represents the fraction of edges connecting j degree nodes and k degree nodes over all the edges in the network with at least one end connected to the k degree nodes. The value of m'_{jk} can be computed from the edge correlation matrix M as

$$m'_{jk} = \frac{m_{jk}}{\sum_{j=0}^{\infty} m_{jk}} = \frac{m_{jk}}{kp_k} \sum_i ip_i \tag{4.38}$$

where $\sum_{j=0}^{\infty} m_{jk}$ denotes the fraction of edge tips connected to k degree nodes in the network and may be expressed as

$$\sum_{j=0}^{\infty} m_{jk} = \frac{kp_k}{\sum_i ip_i} \tag{4.39}$$

Similar to section 4.2, we can say that the number of edge tips connected to the j

degree nodes of set S whose other end is connected to the k degree node of set R becomes $m'_{jk} k n_k f_k (1-f_j)$. This helps us to derive the total number of edges whose one end is connected to a j degree node in set S and the other end is connected to any node in the set R, which can be expressed as

$$E_j = \sum_{k=0}^{\infty} m'_{jk}\, k\, n_k\, f_k\, (1-f_j) \tag{4.40}$$

Due to the presence of degree correlation, the probability that a surviving node of set S loses one link due to the removal of $E(=\sum_{i=0}^{k_M} E_i)$ edges is not constant (as ϕ in Eq. 4.33). Moreover, the probability that a survived node loses one link depends upon the degree (j) of the survived node. Hence, the probability ϕ_j of finding an edge running between a j degree node in the surviving set S and any node of the other set R can be expressed as

$$\phi_j = \frac{E_j}{j n_j (1-f_j)} \tag{4.41}$$

Here ϕ_j signifies the probability that a j degree node loses one link due the removal of E edges.

Finally, using the concept of Eq. (4.4) and from the Eqs. (4.41) and (4.3), the expression of the deformed degree distribution p'_k can be expressed in binomial distribution form

$$p'_k = \sum_{q=k}^{\infty} \begin{pmatrix} q \\ k \end{pmatrix} \phi_q^{q-k} (1-\phi_q)^k \, p_q^s \,. \tag{4.42}$$

where the probability p_q^s of finding a node with degree q in the surviving subset S (before removal of the E edges) is given by Eq. (4.3) of section 4.1.1.

Random failure as a special case

In case of random failure attack the probability of attack on every node is same i.e. $f_j = f_k = f\,(constant)$. Therefore we can express E_j, which is the total number of edges whose one end is connected to a j degree node in set S and the other end is connected to any node in the set R, as the following:

$$E_j = f(1-f)\sum_{k=0}^{\infty} m'_{jk}\, k\, n_k \tag{4.43}$$

Using Eq. (4.38), Eq. (4.43) and (4.39) the expression for E_j reduces to

$$E_j = f(1-f)N\sum_{i=0}^{\infty} ip_i \sum_{k=0}^{\infty} m_{jk} = f(1-f)Njp_j \tag{4.44}$$

We substitute the expression for E_j obtained from Eq. (4.44) in Eq. (4.41) and find

$$\phi_j = \frac{f(1-f)Njp_j}{jn_j(1-f)} = f \tag{4.45}$$

Hence in case of random failure

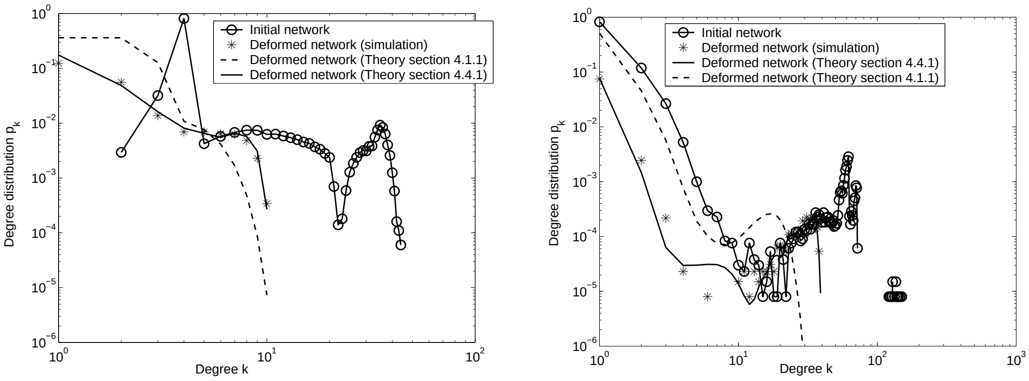
$$\phi = \phi_j = f(constant)\ independent\ of\ any\ degree\ j. \tag{4.46}$$

Substituting the value of $\phi = f$ in Eq. (4.4) and the value $\phi_q = f$ in Eq. (4.42) we find that

$$p'_k(Uncorrelated) = p'_k(Correlated) \tag{4.47}$$

$$= \sum_{q=k}^{\infty} \begin{pmatrix} q \\ k \end{pmatrix} f^{q-k}(1-f)^k\, p^s_q \tag{4.48}$$

The above expression is independent of any correlation parameter. This shows that degree-degree correlation has no role to play in case of random failure. This conclusion confirms the results shown in Figs. 4.12(b) and 4.13(b) where we observe a good agreement of p'_k obtained from the theory and simulation for Gnutella network. However, this does not hold for attacks in correlated networks. Next, we show that our refinement gives better agreement with the simulation results for the attacks on correlated Gnutella networks.



(a) Gnutella A network, correlation coefficient $\alpha = -0.6749$

(b) Gnutella B network, correlation coefficient $\alpha = -0.6318$

Figure 4.15: The impact of deterministic attack upon the degree distribution p_k of the Gnutella network. The figures show that Eq. 4.42 gives far better approximation of the deformed degree distribution than Eq. 4.4

Simulation results on Gnutella Network

We validate the theory developed for correlated network by simulating deterministic attack on 'Gnutella A' and 'Gnutella B' networks. Similar to section 4.3.1, we simulate the deterministic attack on the Gnutella networks. In 'Gnutella A' and 'Gnutella B' network, we simulate deterministic attacks by removing all the nodes with degree greater than 10 and 40 respectively. Fig. (4.15) shows the impact of the deterministic attack on the degree distribution of Gnutella network. It can be observed that the deformed degree distribution obtained from Eq. 4.42 for the Gnutella network is in good agreement with simulation results. We find that the average degree of the 'Gnutella A' and 'Gnutella B' networks obtained from simulation ($Avg\_deg^A_{sim} = 0.4858$ and $Avg\_deg^B_{sim} = 0.1608$) are quite close to the theoretically calculated values using Eq. 4.42 ($Avg\_deg^A_{theory} = 0.4739$ and $Avg\_deg^B_{theory} = 0.1514$).

4.5 Conclusion

In this chapter, we have developed a more sophisticated framework for stability analysis of superpeer networks against attacks. We have shown that this framework enables us to calculate the degree distribution of the deformed network p'_k after removal of nodes. In addition, the framework enables us to measure stability of small scale network as well as networks exhibiting strong degree-degree correlated mixing. As an application of the framework, we have analyzed the effects of two kinds of attacks namely deterministic attack and degree dependent attack and validated the results through simulation. We have shown that in deterministic attack, the increase in peer degree may be detrimental in some cases. Our framework has also revealed that the degree dependent attack provides us a more generalized attack strategy where various situations can be generated only by changing the attack parameter γ. This attack parameter γ also signifies the amount of topological information available to the attacker to breakdown the network. We have observed that increase in γ makes the attack efficient by reducing the percolation threshold. However, beyond a threshold limit, this information does not help the attackers in a significant manner. We have presented a comparative study of our theoretical analysis with real world Gnutella network. The results have shown that degree degree correlation present in Gnutella exhibits a disparity in p'_k in case of attack however the disparity is not seen in case of random failure. We have suitably modified our framework to include the degree-degree correlation factor in consideration. It is important to note that, the stability condition stated in Eq. (4.5) [128] is not applicable for degree-degree correlated network [128]. Hence, in this work we do not derive the percolation threshold of degree correlated network; rather we focus on the accurate calculation of p'_k through a generalized framework. Since degree distribution p'_k is the main ingredient for the stability condition of correlated networks [67], we claim that our work makes a significant contribution towards the understanding stability of generalized network.

In Chapters 3 and 4, we have analyzed the stability of some 'existing' superpeer networks against peer churn and attacks. However, superpeer networks are generally growing networks that continuously evolve with the addition of new peers as well as realignment of peers. Hence, the formation or emergence of superpeer network due to various node and link dynamics is another interesting research problem. The next two chapters focus on the various issues related to the emergence of superpeer networks due to joining and leaving of nodes, rewiring of links etc.

Chapter 5

Emergence of superpeer networks in face of bootstrapping protocols

5.1 Introduction

Superpeer network is formed mainly as a result of the bootstrapping or joining protocol followed by incoming peers. Some other factors like peer churn, rewiring of links also play a major role in the network formation. The superpeer networks emerged following these node and link dynamics exhibit two regimes or ‘bimodality’ in their degree distribution; one regime consists of the large number of low degree peer nodes while the other consists of the small number of high degree superpeers [113]. The emergence of bimodal network due to the node and link dynamics is an *interesting* observation, a rigorous analysis need to be done to understand it. Moreover the performance of the superpeer networks mainly depends upon the topological properties of the emerging networks [20, 139, 144, 170] like network diameter, amount of superpeers in the network, peer-superpeer ratio etc. The analysis will help in regulating these topological properties and subsequently improving the performance of various p2p services will prove to be an *useful* step for p2p research community. In this chapter, we understand the emergence of superpeer networks due to bootstrapping of the incoming nodes and analyze the impact of various nodal parameters on the QoS of different p2p services. In the next chapter, we extend the formalism to include the peer churn and link rewiring and analyze their impact on the various topological properties of the network.

We develop a theoretical framework to explain the appearance of superpeer networks due to the execution of peer servents like limewire, mutella etc [3, 82]. The bootstrapping protocols run by these servents select some 'good' online nodes that are already part of the network and send connection requests to them [82]. We model the bootstrapping protocols by the preferential attachment rule where the probability of joining of an incoming peer to an online node is proportional to the 'goodness' of the online node. 'Goodness' of a peer can be characterized by the *node property* (later quantified as node weight) like amount of resource, processing power, storage space etc that a particular peer possesses [90] as well as its *current degree.* Beyond this, we identify that in p2p networks, bandwidth of a node is finite and restricts its maximum degree (*cutoff degree*). A node, after reaching its maximum degree, rejects any further connection requests from incoming peers. In this chapter, although the basic methodology of preferential attachment is followed, however unlike popular power law, there is an emergence of bimodal degree distribution. We show that the interplay of finite bandwidth with node property play a key role in the emergence of bimodal network [111].
Through suitable mathematical treatment on the framework, we calculate the amount of superpeers in the network, the impact of different parameters like resource, processing power etc on the superpeer-peer ratio etc. As a practical application, we show that our formalism (with a small modification) can almost accurately explain the topological structure of the Gnutella network [65], obtained from the real data taken in 2004 [1]. We believe that this understanding may further help network engineers to appropriately tune the servent programs for improving the p2p services like minimizing search time, fast downloading of files etc.

The outline of the chapter is as follows. In section 5.2, we state and model the bootstrapping protocol followed by peer servents. Section 5.3 proposes a formal framework considering that all the peers join with fixed cutoff degree. In section 5.4, we generalize the theory for the case where different peers join the network with variable cutoff degrees. In light of the framework developed, an empirical analysis of the global nature of the Gnutella 0.6 network is provided in section 5.5. Some suggestions to the network engineers in order to improve the p2p services is provided in section 5.6 after which we conclude this chapter.

5.2 Bootstrapping protocols

In this section, we illustrate and model the bootstrapping protocols that are executed by different servent programs [32]. Servents like limewire and gnucleus maintain a list of 'good' hosts in the GWebCache and give priority to them during connection initiation [82]. We model bootstrapping protocols through node attachment rules where probability of attachment of the incoming peer to an online node is proportional to the node property (weight) and current degree of the online node. The generalized bootstrapping protocol is stated below. The cutoff degree $k_c(i)$ is same for all peers i in the analysis of section 5.3 while it is varied in section 5.4.

```
Input: Nodes, where each node $i$ comes with individual node weight $w_i$ and a
       cutoff degree $k_c(i)$
Output: Network emerged due to joining of the nodes
foreach Incoming node $i$ do
    Node $i$ preferentially chooses $m'$ ($m' > m$) online nodes based on their
    weights and degrees
    while $m$ online nodes are not connected with $i$ do
        $j$ = select an online node among the chosen $m'$ nodes
        Node $i$ sends the connection request to $j$
        if degree(j)$<$ $k_c(j)$ then
            Node $i$ connects with node $j$
        end
        else
            Node $j$ rejects the connection request
        end
    end
end
```

5.3 Development of analytical framework: peers joining with fixed bandwidth

In this section, we develop the analytical framework following the concept of rate equations [11]. We assume that each incoming peer joins the network at any timestep n with some node weight and connects to m online nodes in the network following the bootstrapping protocol. The minimum and maximum weight of a node in the network can be w_{min} and w_{max} respectively. The probability of attachment of the incoming peer to an online node is proportional to the weight and current degree of the online node. The probability that an incoming peer has weight w_i is f_{w_i} and all the nodes have some fixed cutoff degree k_c. Any node upon reaching the degree k_c rejects any further connection request from the incoming peer.

We introduce the term set_{w_i} to denote the set of nodes in the network with weight w_i. Initially we intend to compute p_{k,w_i}, the fraction of k degree nodes in set_{w_i} and then sum it over all sets (weights) to find degree distribution p_k. These values of p_{k,w_i} can be computed by observing the shift in the number of k degree nodes to $k+1$ degree nodes as well as $k-1$ degree nodes to k degree nodes due to the attachment of a new node at timestep n. Let the fraction of nodes in set_{w_i} having degree k at some timestep n be p_{k,n,w_i}, then the total number of k degree nodes in set_{w_i} before addition of a new node is $nf_{w_i}p_{k,n,w_i}$ and after addition of the node becomes $(n+1)f_{w_i}p_{k,n+1,w_i}$. Hence, the change in the number of k degree nodes in set_{w_i} between the timesteps n and $n+1$ becomes

$$\Delta n_{k,w_i} = (n+1)f_{w_i}p_{k,n+1,w_i} - nf_{w_i}p_{k,n,w_i} \tag{5.1}$$

We formulate rate equations depicting these changes for some arbitrary set_{w_i}. By solving those rate equations, we calculate p_{k,w_i} and subsequently the degree distribution p_k (fraction of nodes having degree k) of the entire network.

Methodology

In order to write the rate equations [11], we need to know the attachment probability A_{w_i} that an online node x with weight w_i (i.e. in set_{w_i}) will receive a new link from the incoming peer. The probability that an online node will receive an incoming link is proportional to the node weight w_i and its current degree k and can be depicted as

$$\begin{aligned} A_{w_i} &= \frac{w_i f_{w_i} \sum_{k=m}^{k_c-1} k p_{k,w_i}}{\sum_{i'=min}^{max} w_{i'} f_{w_{i'}} \sum_{k_1=m}^{k_c-1} k_1 p_{k_1,w_{i'}}} \\ &= \frac{w_i f_{w_i} m_{w_i} \beta_i}{\sum_{i'=min}^{max} w_{i'} f_{w_{i'}} m_{w_{i'}} \beta_{i'}} \qquad degree(x) < k_c \qquad (5.2) \\ &= 0 \qquad\qquad degree(x) \geq k_c \end{aligned}$$

where $\beta_i = 1 - \frac{k_c p_{k_c,w_i}}{2m_{w_i}}$ (p_{k_c,w_i} is the fraction of nodes in set_{w_i} that have reached their cutoff degree k_c hence stopped accepting new links) implies the fraction of nodes in set_{w_i} capable of accepting new links from the incoming peer and normalizing constant $2m_{w_i} = \sum_{k=m}^{k_c} k p_{k,w_i}$ denotes the average degree of the nodes in set_{w_i}. The numerator of Eq. (5.2) represents the total amount of weight of nodes in set_{w_i} that are allowed to take incoming links. The denominator normalizes the fraction by the total amount of weight of all the nodes in the network that are allowed to take incoming links.

The joining of a new node of degree m at timestep $n+1$ changes the total number of k degree nodes in set_{w_i}. Since all the nodes in the set_{w_i} contain equal weight w_i, the chance of getting a new link for the online nodes depends upon their current degree k and fraction present in the set at that timestep, hence can be expressed as $\frac{k p_{k,n,w_i}}{2m_{w_i}\beta_i}$. The β_i in denominator takes care of the fact that the nodes, that have reached the cutoff degree k_c do not participate in the formation of new link. Due to the joining of a new node of degree m in the network, some k degree nodes in set_{w_i} acquire a new link and become nodes of degree $k+1$. So the amount of decrease in the number of nodes of degree k, ($m \leq k < k_c$) in set_{w_i} due to this outflux is

$$\delta_{k\to(k+1)} = \frac{k p_{k,n,w_i}}{2m_{w_i}\beta_i} \times A_{w_i} m \qquad (5.3)$$

Similarly a fraction of nodes having degree $k-1$ get a new link and move to the degree k. We now write the rate equations in order to formulate the change in the number of k degree nodes in an individual set_{w_i} due to the attachment of a new node of degree m. Three pertinent degree ranges $k = m$, $m < k < k_c$ and $k = k_c$ are taken into consideration.

Rate equation for $k = m$

Since the probability of joining of a node having weight w_i in the network is f_{w_i}, the joining of one new node of degree m on average increases f_{w_i} fraction of m degree nodes in the set_{w_i}. Hence, net change in the number of nodes having degree $k = m$ can be expressed as

$$\begin{aligned}\Delta n_{m,w_i} &= (n+1)f_{w_i}p_{m,n+1,w_i} - nf_{w_i}p_{m,n,w_i} \\ &= f_{w_i} - \frac{mp_{m,n,w_i}}{2m_{w_i}\beta_i} \times A_{w_i}m \end{aligned} \tag{5.4}$$

Assuming the stationary condition for large n, $p_{k,n+1,w_i} = p_{k,n,w_i} = p_{k,w_i}$ [11] we find

$$p_{m,w_i} = \frac{1}{(1+\frac{m}{\alpha_i})} \tag{5.5}$$

where,

$$\alpha_i = \frac{2\sum_{j=min}^{max} w_j f_{w_j} m_{w_j} \beta_j}{w_i m} = \frac{C}{w_i} \tag{5.6}$$

and $C = \frac{2\sum_j w_j f_{w_j} m_{w_j} \beta_j}{m}$ is a constant.

Similarly from Eq. (5.3), **rate equation for** $m < k < k_c$

$$\begin{aligned}\Delta n_{k,w_i} &= (n+1)f_{w_i}p_{k,n+1,w_i} - nf_{w_i}p_{k,n,w_i} \\ &= \left(\frac{(k-1)p_{k-1,n,w_i} - kp_{k,n,w_i}}{2m_{w_i}\beta_i}\right) \times A_{w_i}m \end{aligned} \tag{5.7}$$

Subsequently, the recurrence relation becomes

$$p_{k,w_i} = \frac{(k-1)}{(k+\alpha_i)}p_{k-1,w_i} \tag{5.8}$$

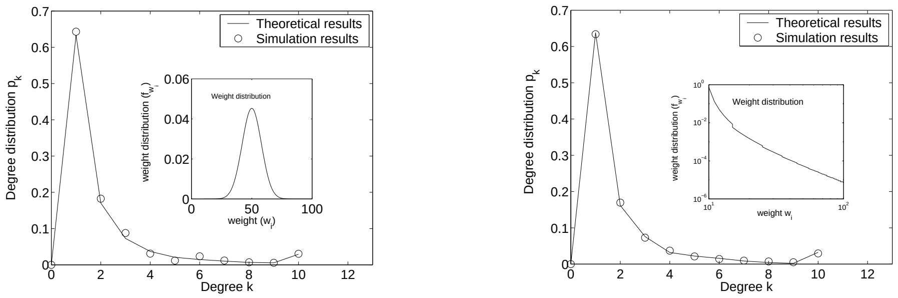
Rate equation for $k = k_c$

Since the nodes having degree k_c are not allowed to take any incoming links, nodes are only accumulated at degree $k = k_c$. Subsequently,

$$\Delta n_{k_c,w_i} = \frac{(k_c-1)p_{k_c-1,n,w_i}}{2m_{w_i}\beta_i} \times A_{w_i}m \tag{5.9}$$

Hence, the corresponding recurrence equation becomes

$$p_{k_c,w_i} = \frac{(k_c-1)}{\alpha_i}p_{k_c-1,w_i} \tag{5.10}$$



(a) Degree distribution of the emerging network. Weight distribution is taken from normal distribution (inset).

(b) Degree distribution of the emerging network. Weight distribution is taken from power law distribution (inset in log-log scale).

Figure 5.1: The plot represents the degree distribution of the network emerged following bootstrapping protocol with fixed cutoff degree $k_c = 10$ and $m = 1$. The nodes join the network with weights taken from normal distribution (mean=50 and standard deviation 8, Fig. 5.1(a)) and power law distribution (exponent=2.5, Fig. 5.1(b)).

Computing the degree distribution

Solving the above stated rate equations, we obtain the degree distribution of the entire network.

$$
\begin{aligned}
p_k &= \sum_{i=min}^{max} p_{k,w_i} f_{w_i} \qquad (5.11)\\
&= \begin{cases}
\sum_{i=min}^{max} \frac{1}{(1+\frac{k}{\alpha_i})} f_{w_i} & k = m \\
\sum_{i=min}^{max} \frac{f_{w_i}}{(1+\frac{m}{\alpha_i})} \times \prod_{j=1}^{k-m} \left(\frac{k-j}{k-j+1+\alpha_i} \right) & m < k < k_c \\
\sum_{i=min}^{max} f_{w_i} \prod_{j=m}^{k-1} \frac{j}{(j+\alpha_i)} & k = k_c
\end{cases}
\end{aligned}
$$

5.3.1 Emergence of superpeer nodes

We are now in the position to theoretically understand the emergence of bimodal distribution as well as the accumulation of superpeer nodes. A closer look at the equations reveals that two modes appear in the degree distribution, one at $k = m$

around which the degree of most of the nodes are concentrated and another at $k = k_c$.
Two conditions need to be satisfied.
a. In order to show the appearance of mode or spike at $k = k_c$, we have to satisfy the condition $p_{k_c} > p_{k_c-1}$ and $p_{k_c} > p_{k_c+1}$.
b. In order to show the modal behavior at $k = m$, we have to satisfy the condition $p_k < p_{k-1}$ for $m \leq k < k_c$. This also confirms that no other modes have emerged in the network.
Fulfilling condition a: First of all, we show that the fraction of nodes having degree k_c, p_{k_c} is greater than p_{k_c-1}. From Eq. (5.11), we find

$$\frac{p_{k_c}}{p_{k_c-1}} = \frac{\sum_{i=min}^{max} p_{k_c,w_i} f_{w_i}}{\sum_{i=min}^{max} p_{k_c-1,w_i} f_{w_i}} = \frac{\sum_i (k_c - 1) x_i}{\sum_i \alpha_i x_i} \tag{5.12}$$

where

$$x_i = \frac{1}{(m + \alpha_i)(m + 1 + \alpha_i)......(k_c - 1 + \alpha_i)}$$

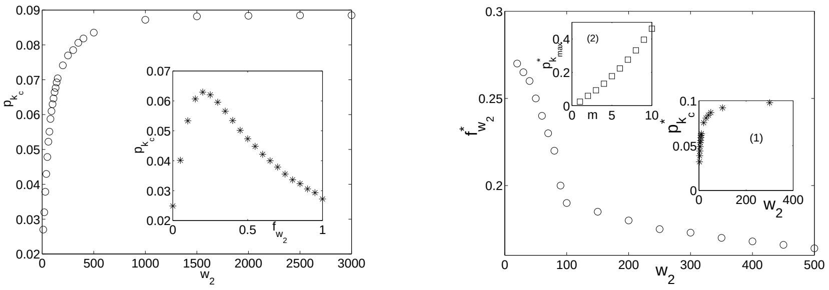
Since $\sum_i m_{w_i} f_{w_i} = m$ and $\beta_i < 1$ therefore $m_{w_i} f_{w_i} \beta_i < m$, hence $\sum_i (k_c - 1) > \sum_i \alpha_i$ as $k_c >> 1$. This confirms $p_{k_c} > p_{k_c-1}$. Secondly, the bootstrapping protocol gives $p_k = 0$ for $k > k_c$. Hence, we conclude the presence of a spike at degree k_c.
Fulfilling condition b: We find for $m \leq k < k_c$, the probability p_k continuously decreases. This can be understood from Eq. (5.8) of the set_{w_i}

$$\frac{p_{k,w_i}}{p_{k-1,w_i}} = \frac{(k-1)}{(k+\alpha_i)} < 1 \tag{5.13}$$

i.e. $p_{k,w_i} < p_{k-1,w_i}$. Hence for the entire network, $p_k < p_{k-1}$. These two observations confirm the presence of two distinct modes in the degree distribution and lead to the emergence of high degree superpeer nodes at degree k_c (Figs. 5.1(a), 5.1(b)). Note that, this feature is independent of the weight distribution f_w.

5.3.2 Simulation results and inference derivation

We validate the theoretically obtained degree distribution (Eq. (5.11)) by simulating the emergence of the network (Fig. 5.1). In these simulations, we follow the exactly same procedure and assumptions that we have considered for theoretical modeling.



(a) The plot illustrates the change in p_{k_c} due to change in w_2 and f_{w_2} for the bimodal weight distribution (simulation results).

(b) Change in $f^*_{w_2}$ due to the increase in w_2. Inset(1) - the corresponding $p^*_{k_c}$ calculated at $f^*_{w_2}$. Inset(2) - $p^*_{k_c max}$ (using $f^*_{w_2}$ and $w_2 \rightarrow \infty$) with m.

Figure 5.2: Fig. 5.2(a) shows the change in p_{k_c} due to change in w_2 and f_{w_2} for the bimodal weight distribution. Inset indicates the presence of optimum f_{w_2} (i.e. $f^*_{w_2}$) at which p_{k_c} becomes maximum ($p^*_{k_c}$). Fig. 5.2(b) shows the change in $f^*_{w_2}$ and $p^*_{k_c}$ due to w_2 (simulation results).

The stochastic simulation set up is as follows. During bootstrapping, each node joins the network with some weight ($10 \leq w \leq 100$) taken from a weight distribution f_w. A 'fitness' value is assigned to each online node based upon its weight and current degree. The incoming new node gets connected with an online node depending upon the 'fitness' of that online node. In our simulation, we consider two different weight distributions, namely normal distribution and power law distribution [90,144]. The total number of nodes in the system is considered to be 5000 and we perform 500 individual realizations and plot the average degree distribution. Fig. 5.1 shows that the agreement between the theoretical and simulation results is exact which validates the correctness of the theoretical model. Figs 5.1(a), 5.1(b) produce the evidence of the emergence of two distinct regions in the degree distribution - the peer and superpeer regions; the accumulation of the superpeer nodes occurs at degree $k_c = 10$. Fig. 5.1 confirms that the weight distribution hardly changes the nature (i.e. bimodalilty) of the degree distribution. In the following, we investigate the influence of different parameters on the amount of superpeers in the network (p_{k_c}). In order to gain more insights, we consider a simple bimodal weight distribution where nodes join with two weights w_1 (low) and w_2 (high) with individual fractions f_{w_1} (high) and f_{w_2} (low) respectively.

Impact of node weight w_2 on p_{k_c}

In order to examine the impact of node weight, we perform the simulation with $w_1 = 10$ and $f_{w_1} = 0.8$. The node weight w_2 is varied from 10 to 3000 and we observe how it affects p_{k_c} (k_c=10). It can be observed from Fig. 5.2(a) that, initial increase in w_2 increases the fraction of superpeer nodes (p_{k_c}) in the network rapidly. However, after a certain threshold, the p_{k_c} stabilizes and further increase in weight does not increase p_{k_c}. Mathematically from Eq. (5.11), as $w_2 \rightarrow \infty$, p_{k_c} becomes

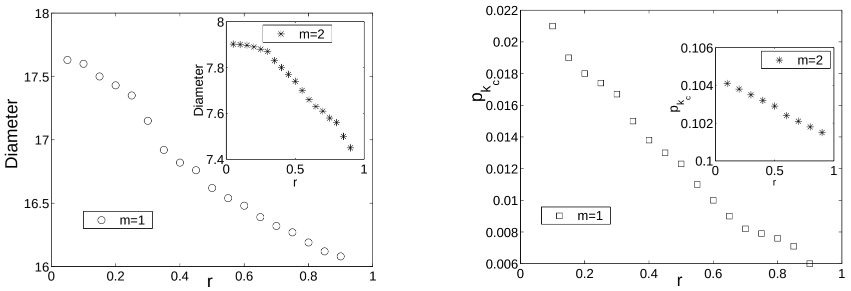
$$\lim_{w_2 \rightarrow \infty} p_{k_c} = f_{w_2} \prod_{j=m}^{k_c-1} \frac{j}{(j + \frac{2}{m} f_{w_2} m_2 \beta_2)} \tag{5.14}$$

and converges to some finite value. Hence, we conclude that after some threshold limit, increase in the node weight w_2 does not increase the amount of superpeers in the network.

Impact of fraction of high weighted nodes (f_{w_2}) on p_{k_c}

In order to observe the impact of f_{w_2} on p_{k_c}, we simulate the bootstrapping protocol for two weights $w_1 = 10$ and $w_2 = 100$ and gradually increase the f_{w_2} (i.e. decrease f_{w_1}). Common intuition is that increase in f_{w_2} in the network should increase p_{k_c} (number of superpeers) as well. However inset of Fig. 5.2(a) shows that the initial increase in f_{w_2} increases p_{k_c}. But after reaching some maximum value ($p^*_{k_c}$), p_{k_c} decreases. We are interested in understanding the reason behind the presence of an optimum f_{w_2} ($f^*_{w_2}$, at which p_{k_c} becomes maximum). This can be understood by looking into the opposite forces performing at two ends (high and low) of $f^*_{w_2}$.

In low f_{w_2} : During the joining of a new node of degree m, the existing nodes in the network acquire the links from the new node and scale their own degrees. Low f_{w_2} (i.e. high f_{w_1}) makes the $w_1 f_{w_1}$ quite significant and subsequently increases A_{w_1} in



(a) The plot illustrates the change in the diameter of the network with the change in bootstrapping protocol (r).

(b) The plot illustrates the change in the amount of superpeers (p_{k_c}) of the network with the change in bootstrapping protocol (r).

Figure 5.3: In Figs 5.3(a) and 5.3(b), r is the fraction of incoming nodes which have joined the network purely based on the degree sequence of the online nodes. The results are obtained through stochastic simulation.

Eq. (5.2). In effect, out of m links of the incoming node, some of them get connected to w_1 weight nodes. However, since w_1 is small, any individual w_1 weighted node rarely becomes capable to reach k_c for contributing to p_{k_c}. But, collectively they restrict the w_2 weighted nodes from taking new links, hence reduce the rate of degree scaling of those nodes. This results in low value of p_{k_c}. **In high f_{w_2}** : However in high f_{w_2}, all the nodes of weight w_2 compete with each other to get the new links. This results in slowdown in the rate of increase of the degrees of the individual w_2 weighted nodes and gradually reduces p_{k_c}. The interaction of these two opposite effects results in the emergence of an optimal $f^*_{w_2}$.

Impact of w_2 on $f^*_{w_2}$

Fig. 5.2(b) shows that the increase in w_2 sharply decreases the $f^*_{w_2}$. Increase in w_2 increases A_{w_2}, hence most of the links of the incoming node get attached to the nodes with high weight w_2 even if f_{w_2} is small. At the same time, low f_{w_2} restricts competition for the incoming links among the w_2 nodes and helps the small fraction of high degree nodes to quickly scale towards the cutoff degree k_c. Inset(1) of Fig. 5.2(b) shows that the interplay of these two factors increases $p^*_{k_c}$ (i.e. p_{k_c} at $f^*_{w_2}$). However, after reaching the saturated w_2, all the incoming links are joined to the w_2 nodes hence further increase in w_2 does not reduce the $f^*_{w_2}$ (or increase $p^*_{k_c}$) much.

Increase in m increases the amount of superpeers:

Eq. (5.14) calculates the maximum amount of superpeers in the network as $w_2 \rightarrow \infty$ for different f_{w_2}. The optimum fraction $f^*_{w_2}$ can be calculated from Eq. (5.14) by taking $\frac{dp_{k_c}}{df_{w_2}} = 0$. Substituting that $f^*_{w_2}$ in Eq. (5.14) gives the maximum possible amount of superpeers $p^*_{k_cmax}$ for a particular m. Inset(2) of Fig. 5.2(b) shows that with the increase in m, the $p^*_{k_cmax}$ increases almost linearly.

Impact of the bootstrapping protocol on p2p services

In this subsection, we investigate the implications of some modifications in the bootstrapping protocols on the various network properties like diameter, amount of superpeers etc. Let us assume that the bootstrapping protocol of the incoming peer can be controlled such that probability of connecting with only high degree online nodes is r and probability of connecting with an online node based upon both its weight and degree is $1 - r$. In simulation, we assume that the weight distribution of the incoming nodes follow power law distribution [144]. Fig. 5.3(a) shows that increasing r slowly decreases the diameter of the network. Reducing the diameter of the network improves the search efficiency of the network [130]. On the other hand, increasing r reduces the amount of superpeers in the network p_{k_c} (Fig 5.3(b)). As the file download latency is primarily dependent on the nature of the neighboring peers, the increase in the amount of superpeers results in fast downloading of files. Hence we conclude that carefully modifying the bootstrapping protocol to sieve appropriate nodes from the GWebCache may improve the p2p services by reducing search latency and improving file download speed etc.

5.4 Development of analytical framework: peers joining with individual/variable bandwidth

In reality, nodes join the network with various bandwidth connections like dial up, ISDN, ADSL, leased line etc. Subsequently, the cutoff degree of individual nodes becomes different from one another. For simplicity, we can assume that there is a fixed number of discrete cutoff degrees each representing a type of connection. We therefore generalize the bootstrapping in the following way. We assume that the probabilities that a node j joins the network with cutoff degree $k_c(j)$ and weight w_j are $q_{k_c(j)}$ and f_{w_j} respectively ($q_{k_c(j)}$ and f_{w_j} are independent). Let every node necessarily have cutoff degree between a specified minimum and maximum, $k_c(min)$ and $k_c(max)$, respectively. Similar to the section 5.3, the probability that an online node of weight w_i (i.e. in set_{w_i}) receives a new link from the incoming peer is

$$\begin{aligned} \widehat{A}_{w_i} &= \frac{w_i f_{w_i}(\sum_{k=m}^{k_{min}-1} kp_{k,w_i} + \sum_{k=k_{min}}^{k_{max}} kp_{k,w_i}S_{k,w_i})}{\sum_{i'=min}^{max} w_{i'}f_{w_{i'}}(\sum_{k=m}^{k_{min}-1} kp_{k,w_{i'}} + \sum_{k=k_{min}}^{k_{max}} kp_{k,w_{i'}}S_{k,w_{i'}})} \\ &= \frac{w_i f_{w_i} m_{w_i} \widehat{\beta}_i}{\sum_{i'=min}^{max} w_{i'} f_{w_{i'}} m_{w_{i'}} \widehat{\beta}_{i'}} \end{aligned} \tag{5.15}$$

where

$$\widehat{\beta}_i = 1 - \frac{\sum_{k=k_c(min)}^{k_c(max)} (1 - S_{k,w_i}) kp_{k,w_i}}{2m_{w_i}} \tag{5.16}$$

implies the fraction of nodes in set_{w_i} capable of accepting new links from the incoming peer. Here S_{k,w_i} is the fraction of k degree nodes in set_{w_i} whose cutoff degree is greater than k and hence are still capable of taking incoming connections. We calculate the exact expression for S_{k,w_i} later in this section.

Similar to the section 5.3, we formulate the rate equations to characterize joining of an incoming node of degree m. Based on the behavior of S_{k,w_i}, the formulation of rate equations and subsequently the computation of degree distribution need to be done in two parts; nodes with degree $m \leq k < k_c(min)$ in part A and nodes with degree $k_c(min) \leq k \leq k_c(max)$ in part B.

Part A : Dynamics analysis for $m \leq k < k_c(min)$

In this case, none of the nodes has reached its cutoff degree. Hence S_{k,w_i} trivially becomes 1 and the rate equations for $m \leq k < k_c(min)$ are similar to the Eqs. (5.4) and (5.7).

Part B : Dynamics analysis for $k_c(min) \leq k \leq k_c(max)$

An important difference between part B and part A is that, at each k ($k_c(min) \leq k \leq k_c(max)$), a fraction of nodes reach to their cutoff degree and stop accepting further links from the incoming nodes. So the calculation of S_{k,w_i} becomes nontrivial and their values play a major role in formulating the rate equations. We start our analysis with the nodes having smallest cutoff degree $k = k_c(min)$.

(B_1) Calculation for $k = k_c(min)$

We defined earlier that S_{k,w_i} is the fraction of nodes having degree $k = k_c(min)$ in the set_{w_i} that have not reached their cutoff degree and are still capable of taking incoming links. Hence similar to Eq. (5.3), $\frac{kp_{k,w_i}}{2m_{w_i}\widehat{\beta}_i}\widehat{A}_{w_i}mS_{k,w_i}$ number of nodes can move from degree $k_c(min)$ to $k_c(min)+1$ and leave the $k_c(min)$ set. On the other hand, similar to Eq. (5.3), the mean number of nodes with degree $k-1$ that accepts new links and moves to degree k becomes $\frac{(k-1)p_{k-1,w_i}}{2m_{w_i}\widehat{\beta}_i}\widehat{A}_{w_i}m$. The net change in the number of nodes having degree k (for $k = k_c(min)$) due to the attachment of a new node is

$$\Delta n_{k,w_i} = \frac{((k-1)p_{k-1,w_i} - kp_{k,w_i}S_{k,w_i})}{2m_{w_i}\widehat{\beta}_i} \times \widehat{A}_{w_i}m \tag{5.17}$$

Calculation of S_{k,w_i} for $k = k_c(min)$

The mean number of nodes of degree $(k-1)$ that acquires the new links from the incoming node and moves from degree $k-1$ to degree k is $\widehat{\delta}^{jo}_{(k-1)\to k} = \frac{(k-1)p_{k-1,w_i}}{2m_{w_i}\widehat{\beta}_i}\widehat{A}_{w_i}m$. As q_k is the probability that a node joins the network with cutoff degree $k = k_c(min)$, hence $\widehat{\delta}^{jo}_{(k-1)\to k} \times \frac{q_k}{\sum_{k'=k}^{k_c(max)} q_{k'}}$ specifies the number of nodes that moves from degree $k-1$ to k and also reaches its cutoff degree $k = k_c(min)$. If the fraction of k degree nodes in set_{w_i} is p_{k,w_i}, then the fraction of nodes reaching the cutoff degree k can be normalized as

$$1 - S_{k,w_i} = \frac{\frac{(k-1)p_{k-1,w_i}}{2m_{w_i}\widehat{\beta}_i}\widehat{A}_{w_i}mq^*_k}{p_{k,w_i}} \Rightarrow S_{k=k_c(min),w_i} = 1 - \frac{\frac{(k-1)p_{k-1,w_i}}{2m_{w_i}\widehat{\beta}_i}\widehat{A}_{w_i}mq^*_k}{p_{k,w_i}} \tag{5.18}$$

where $q_k^* = \frac{q_k}{\sum_{k'=k}^{k_c(max)} q_{k'}}$. Substituting the value of S_{k,w_i} in Eq. (5.17) and rearranging p_{k,w_i}, we get

$$p_{k,w_i} = \frac{(k-1)}{(k+\widehat{\alpha}_i)}\left(1+\frac{kf_{w_i}q_k^*}{\widehat{\alpha}_i}\right)p_{k-1,w_i} \tag{5.19}$$

where

$$\widehat{\alpha}_i = \frac{2\sum_{j=min}^{max} w_j f_{w_j} m_{w_j}\widehat{\beta}_j}{w_j m} = \frac{\widehat{C}}{w_j} \tag{5.20}$$

and $\widehat{C} = \frac{2\sum_j w_j f_{w_j} m_{w_j}\widehat{\beta}_j}{m}$ is a constant.

(B_2) Calculation for $k = k_c(min) + 1$

This case differs from the previous ($k = k_c(min)$) in one aspect - unlike previous case, only $S_{k_c(min),w_i}$ (i.e. S_{k-1,w_i}) fraction of $(k-1)$ degree nodes can accept incoming links and change their degree to k. Hence, in this case the rate equation becomes

$$\Delta n_{k,w_i} = \frac{((k-1)p_{k-1,w_i}S_{k-1,w_i} - kp_{k,w_i}S_{k,w_i})}{2m_{w_i}\widehat{\beta}_i} \times \widehat{A}_{w_i}m \tag{5.21}$$

Calculation of S_{k,w_i} for $k = k_c(min) + 1$

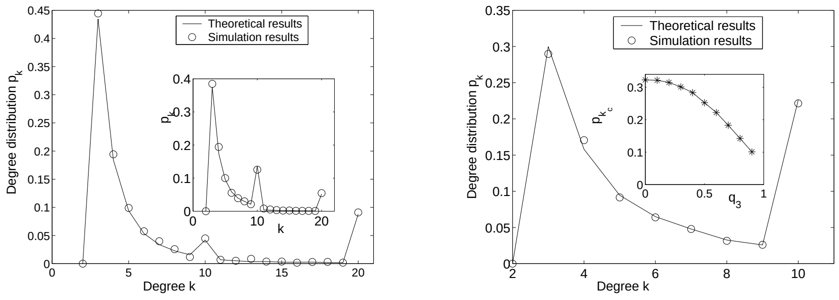
The mean number of nodes of degree $(k-1)$ that acquires the new links from the incoming node and moves to degree k is $\widehat{\delta}^{jo}_{(k-1)\to k} = \frac{(k-1)p_{k-1,w_i}}{2m_{w_i}\beta_i} \times \widehat{A}_{w_i} m S_{k-1,w_i}$. As q_k is the probability that a node joins the network with cutoff degree $k = k_c(min)+1$, hence $\widehat{\delta}^{jo}_{(k-1)\to k} \times q_k^*$ specifies the number of nodes that reaches the cutoff $k = k_c(min) + 1$. With proper normalization, we obtain

$$S_{k=k_c(min)+1,w_i} = 1 - \frac{\frac{(k-1)p_{k-1,w_i}}{2m_{w_i}\beta_i} \times \widehat{A}_{w_i} m S_{k-1,w_i} q_k^*}{p_k} \tag{5.22}$$

Substituting the values of S_{k,w_i}, S_{k-1,w_i} in Eq. (5.21), we get

$$p_{k,w_i} = \frac{(k-1)}{(k+\widehat{\alpha}_i)}\left(1+\frac{kf_{w_i}q_k^*}{\widehat{\alpha}_i}\right)\left(p_{k-1,w_i} - \frac{(k-2)p_{k-2,w_i}f_{w_i}q_{k-1}^*}{\widehat{\alpha}_i}\right) \tag{5.23}$$

Generalization : Continuing the calculations for $k_c(min) < k \leq k_c(max)$, we obtain



(a) Degree distribution of the emerging network in case 1 and case 2 (inset).

(b) Degree distribution of the emerging network in case 3. Inset shows the impact of lower cutoff degrees on p_{k_c}.

Figure 5.4: Case 1: fractions of nodes joined with cutoff degrees $3, 10$ and 20 are $0.5, 0.1$ and 0.4 respectively. Case 2: fractions of nodes joined with cutoff degrees $3, 10$ and 20 are $0.5, 0.3$ and 0.2 respectively (Inset). Fig. 5.4(b) shows case 3 where 50% nodes joined with cutoff degree 3 and rest 50% joined with cutoff degree 10. Inset shows the change in p_{k_c} (at $k_c = 10$) in the network due to the increase in q_3 (the fraction of nodes with cutoff degree 3).

the generalized equation

$$p_{k,w_i} = \frac{(k-1)}{(k+\widehat{\alpha}_i)}\left(1+\frac{kf_{w_i}q_k^*}{\widehat{\alpha}_i}\right) \left(p_{k-1,w_i} + \sum_{j=1}^{k-k_c(min)} (-1)^j \prod_{t=1}^{j} \frac{(k-t-1)p_{k-t-1,w_i}f_{w_i}q_{k-t}^*}{\widehat{\alpha}_i}\right) \tag{5.24}$$

The degree distribution of the entire network p_k is calculated by summing up p_{k,w_i} over all w_i's, i.e. $p_k = \sum_{i'=min}^{max} p_{k,w_{i'}} f_{w_{i'}}$.

5.4.1 Simulation results and inference derivation

The trend which emerges behind such complicated equations is next explained through analysis and illustration.

Emergence of superpeer nodes

Fig. 5.4(a) shows that if peers join with (say) v different cutoff degrees, the degree distribution of the network shows upto v (say $\widehat{v}$) spikes. We observe that the exact value of $\widehat{v}$ typically depends upon the fraction of nodes joining the network with a particular cutoff degree. Theoretically probing into the equations gives a better idea. Let us assume that the nodes join the network with v distinct and far apart (i.e. $k_c(a_{j+1}) > k_c(a_j) + 1$) bandwidths with cutoff degrees being $k_c(a_1)$, $k_c(a_2)$, $k_c(a_3)$ ... $k_c(a_v)$ respectively where $k_c(a_1)$ is the smallest cutoff and $k_c(a_v)$ is the highest one. Fraction of nodes joining with cutoff degree $k_c(a_i)$ is $q_{k_c(a_i)}$ for $1 \leq i \leq v$.
Condition: $p_{k_c(a_i)-1} < p_{k_c(a_i)} > p_{k_c(a_i)+1}$ confirms the appearance of spike at degree $k_c(a_i)$. The analysis follows. Calculating $p_{k_c(a_i)+1,w_i}$ and $p_{k_c(a_i),w_i}$ and eliminating $p_{k_c(a_i)-1,w_i}$, we get $\frac{p_{k_c(a_i)+1,w_i}}{p_{k_c(a_i),w_i}} < 1$, hence for the entire network, $p_{k_c(a_i)+1} < p_{k_c(a_i)}$; that is the fraction of nodes having degree one more than some cutoff degree (say $k_c(a_i) + 1$) is less than the fraction of nodes at that cutoff degree (say $k_c(a_i)$).
Similarly, from Eq. (5.19) we find

$$\frac{p_{k_c(a_i),w_i}}{p_{k_c(a_i)-1,w_i}} = \frac{(k_c(a_i) - 1)}{(k_c(a_i) + \widehat{\alpha}_i)} \left[1 + \frac{k_c(a_i) f_{w_i} q^*_{k_c(a_i)}}{\widehat{\alpha}_i}\right] \tag{5.25}$$

In order to satisfy $p_{k_c(a_i)} > p_{k_c(a_i)-1}$, we find that if $q_{k_c(a_i)}$ (the fraction of nodes joined the network with cutoff degree $k_c(a_i)$) is above a threshold level, then only a mode or spike appears at degree $k_c(a_i)$.

Simulation results

In order to validate our theoretical framework, we simulate the bootstrapping protocol where nodes join the network with variable cutoff degrees. We consider that the weight distribution (f_w) of the incoming nodes follows power law distribution (with exponent=2.5) [144, 149] and the nodes can have 3 different cutoff degrees $3, 10$ and 20. At the time of joining, each node establishes connections with 3 online nodes in the network i.e. $m = 3$. We assume that the 50% of nodes join through (say) dial up connection having cutoff degrees 3. Rest 10% of nodes join through (say) ISDN connection with cutoff degree 10 and 40% through (say) leased line connection with cutoff degree 20. We assume that all the nodes having degree ≥ 10 can be considered as superpeer nodes [170]. The total number of nodes in the simulation system is 5000 and 500 different realizations are performed. Fig. 5.4(a) shows that the agreement between the theoretical model (Eq. (5.24)) and simulation is exact.

Measuring the amount of superpeers in the network

Fig. 5.4(a) shows that in case 1, total fraction of superpeer nodes (i.e. degree ≥ 10) in the network is 0.1472. On the other hand, if the fraction of nodes joining with cutoff degrees $3, 10$ and 20 is $0.5, 0.3$ and 0.2 respectively (inset of Fig. 5.4(a), referred as case 2), the fraction of superpeers in the network becomes 0.2158. If 50% of nodes join with cutoff 3 and rest 50% joins with a cutoff 10, the total fraction of superpeers in the network becomes 0.2361 (Fig. 5.4(b), referred as case 3). Hence our results show that instead of joining the network through multiple high bandwidth connections, using a single bandwidth is optimal for the emergence of highest amount of superpeers in the network.

Effect of low cutoff degrees: In Fig. 5.4(b) (inset), we consider a situation where the nodes join with two cutoff degrees; q_3 fraction of nodes join with cutoff degree 3 and rest $q_{10} = (1 - q_3)$ fraction of nodes join with higher cutoff degree 10. In the idealistic case, when all the nodes join with cutoff degree 10 (i.e. $q_3 = 0$), the amount of superpeers in the network would be maximum ($p_{k_c} = 0.32$). The amount would decrease as some nodes with lower bandwidth hence lower cutoff degree (here 3) joins the network. The plot in Fig. 5.4(b) (inset) shows the rate at which p_{k_c} ($k_c = 10$) decreases. We find that the fraction of superpeer nodes hardly changes as long as percentage of nodes with cutoff degree 3 are less than 20%.

5.5 Case study with Gnutella network

We simulate Gnutella network following the snapshot obtained from the Multimedia & Internetworking Research Group, University of Oregon, USA [1]. The snapshot is collected by the research group during September 2004 and the size of the network

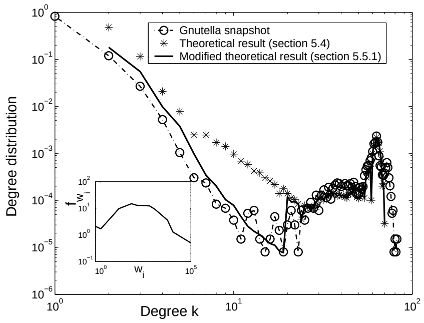


Figure 5.5: Degree distribution of Gnutella network taken from the topological snapshot [1]. The inset shows the weight distribution of the incoming nodes [20]. We assume that the weight of a node can be determined by the amount of shared files it possesses (indicates the shared resource) and inverse of node latency (indicates the node's processing power). The cumulative distribution of the amount of shared files and latency of the Gnutella peers are available in [20]. We take a joint probability distribution of these two parameters in order to get the weight distribution (inset). The figure illustrates the comparative study between the real world Gnutella networks [1] and our theoretical model.

simulated from the snapshot is of $1,31,869$ nodes. In order to verify whether the degree distribution of Gnutella can be explained through the developed framework, we theoretically compute the degree distribution of the emerging network (from section 5.4) by taking the weights from the weight distribution of the inset of Fig. 5.5 [20]. During connection initiation, most of the servents initially connect to multiple online peers [82], therefore we keep $m = 2$. The probability $q_{k_c(j)}$ of joining of a node j with cutoff degree $k_c(j)$ is adjusted accordingly to fit the calculated degree distribution close to the Gnutella network. As can be seen from Fig. 5.5, our theoretical model can mimic the degree distribution of Gnutella network with reasonable accuracy, however there are some deviations. Although the higher degree nodes match almost exactly with theory, the amount of small degree nodes in Gnutella is less than the theoretically calculated p_k. The possible reason is, due to the finite size of the web cache, the GWebCache is totally populated by the high degree nodes in the network. Henceforth, the peers having low degree do not receive any connection from the incoming node. Thus most of the low degree peer nodes remain with the low degree and subsequently the amount of low degree nodes in Gnutella network becomes lower than theoretically calculated value. Next we address the finite size web cache issue and modify the formalism accordingly.

5.5.1 Modifying the formalism with finite size WebCache

In order to model the finite size web cache, we assume that the nodes having degree greater than $m'(m' < k_c(min))$ be always present in the web cache (with probability 1). However, the probability of getting a node in the webcache having degree k, such that $m \leq k \leq m'$ is γ. We suitably modify the rate equations described in Eqs. (5.4) and (5.7) to incorporate these assumptions. It is important to note that, as $m' < k_c(min)$, these changes may only affect the calculations of the part A of section 5.4.

Similar to Eq. (5.3), the average number of m degree nodes in the webcache acquiring links from the incoming node becomes

$$\gamma \frac{m p_{m,n,w_i}}{2 m_{w_i} \beta_i} \times A_{w_i} m \tag{5.26}$$

Hence modified rate equation for $k = m$

$$\Delta n_{m,w_i} = f_{w_i} - \gamma \frac{m p_{m,n,w_i}}{2 m_{w_i} \beta_i} \times A_{w_i} m \tag{5.27}$$

Similarly, the rate equation for $m \leq k \leq m'$

$$\Delta n_{m,w_i} = \gamma \left(\frac{(k-1) p_{k-1,n+1,w_i} - k p_{k,n,w_i}}{2 m_{w_i} \beta_i} \right) \times A_{w_i} m$$

Hence the modified degree distribution becomes

$$p_k = \sum_{i=min}^{max} p_{k,w_i} f_{w_i} \tag{5.28}$$

$$= \begin{cases} \sum_{i=min}^{max} \frac{1}{(1+\frac{\gamma k}{\alpha_i})} f_{w_i} & k = m \\ \sum_{i=min}^{max} \frac{f_{w_i}\gamma^{k-m}}{(1+\frac{\gamma m}{\alpha_i})} \times \prod_{j=1}^{k-m} \left(\frac{k-j}{\gamma(k-j+1)+\alpha_i}\right) & m < k < m' \\ \sum_{i=min}^{max} \frac{f_{w_i}\gamma^{k-m}}{(1+\frac{\gamma m}{\alpha_i})} \times \prod_{j=1}^{k-m} \left(\frac{k-j}{k-j+1+\alpha_i}\right) & k = m' + 1 \end{cases}$$

Calculation of p_k for the nodes having degree $k > m' + 1$ remains same as sections 5.3 and 5.4. We plot the modified equations (Eq. (5.28)) in Fig 5.5 with $\gamma = 0.37$ and $m' = 18$ which fits the *Gnutella snapshot almost perfectly.*

5.6 Conclusion and design guidelines to the network engineers

The work done in this chapter brings forward an important message that preferential attachment may also result in a bimodal degree distribution which superpeer topologies exhibit. This happens when preferential attachment takes into consideration three features simultaneously; the node weight (quantifies the amount of resource, processing power, storage space etc.), current degree and the available bandwidth. The developed formalism points to the fact that accurate computation of the degree distribution of a network is possible (as shown in the Fig. 5.5 for Gnutella) based on the bootstrapping protocol and the information about the nature of web cache.

The developed formalism and rigorous analysis lead to some suggestions which if used, would result in minimal change in the present servent implementations, however may lead to a quantum jump in performance. Specifically two areas of servent program - bootstrapping protocol and GWebCache updation can be improved. (a) **Bootstrapping :** The bootstrapping protocols can be properly modified to control the amount of superpeers in the network. Section 5.4.1 shows that instead of joining the network with different bandwidth levels, using a few (or single) cutoff degrees is optimal for the emergence of high amount of superpeers in the network. In Gnutella, different nodes in the network join with their individual bandwidth (or cutoff degree) that varies widely (dial up connection, ADSL, LAN). However, the bootstrapping protocol can be properly designed to restrict the maximum degree of the individual nodes to a few small number. This measure will result in the higher presence of superpeer nodes in the network (Fig. 5.4(b)) and subsequently facilitate proper load balancing in the system. (b) **Updation of GWebCache :** In addition to that, rigorous analysis of our formalism leads to some suggestions to the network engineers regarding the updation of GWebCache. Two important results have been reported (a) high weighted node can increase the fraction of superpeers only upto a level (section 5.3.2) (b) presence of too many high weighted nodes may be detrimental (section 5.3.2). GWebCache is periodically populated by the online peers/superpeers nodes based on the specific servent implementation [82]. Hence instead of blindly updating the GWebCache with 'high weighted' nodes, updation techniques which properly balance nodes' weight and degree can be undertaken.

In this chapter, we have focused on the formation of superpeer network only due to the bootstrapping of joining nodes. However, in addition to the bootstrapping, the frequent departure of the online peers and relinking of the existing connections play a major role in the topology formation of the network. In the next chapter, we include peer churn and rewiring of links in our formalism and analyze their effects on the topology (like amount of superpeers, largest connected component, network diameter etc) as well as on the various p2p services. The damage in the network connectivity caused by peer churn and repairing activity initiated by the rewiring of links is rigorously analyzed.

Chapter 6

Emergence of superpeer networks in face of churn and link rewiring

6.1 Introduction

In Chapter 5, we have investigated the reason behind the emergence of superpeer networks by modeling bootstrapping as a preferential attachment process. We have made the simplified assumption that the network does not undergo any node churn or rewiring, hence we only deal with the joining of incoming nodes through bootstrapping protocols. But in reality, any p2p network is highly dynamic with nodes and links continuously undergoing churn/reformation [160]. Hence any understanding, even qualitative, of the topology remains grossly incomplete without considering these two dynamics. In this chapter, we extend the formalism, developed in Chapter 5 to include churn, rewiring along with bootstrapping [7]. In the last chapter, we have assumed that the 'goodness' of a node is proportional to the node weight and current node degree. And accordingly, we have performed a detailed study of the impact of node weights on the accumulation of the superpeer nodes in the network. Since peer churn and link rewiring is the primary focus of this chapter, to keep calculation simple, we characterize the 'goodness' of a node only by its current degree. We have seen that this assumption does not affect the generality of our formalism and if necessary, the 'weight' parameter may be easily included in our framework.

The formalism developed in this chapter unfolds the bounds till which the qualitative nature of superpeers is preserved against churn. More importantly, it gives concrete idea of many of the topological parameters like amount of superpeers, component size, network diameter etc. As we have observed in Chapter 5, the quality of service enjoyed by a particular peer is mainly determined by the nature of its neighboring nodes, favorably towards superpeer nodes. Subsequently, the high amount of superpeers in the network improves the overall QoS. Side by side, the impact of churn and rewiring on the network connectivity may be indicated by the size of the largest connected component in the network whereby one can understand the extent of communication possible among peers. Similarly the network diameter directly affects the search efficiency of the networks [130]. Analyzing the influence of peer churn and link rewiring upon all these topological properties is the primary focus of this chapter.
The rest of the chapter is organized as follows. In section 6.3, we assume that rewiring process is not present hence only consider that nodes join through bootstrapping and leave the network through peer churn. We consider that all the peers join the network with fixed cutoff degree. In this section, we take a special case to show the network behavior without churn. These results become useful next, when we investigate the impact of churn on the superpeer network. In section 6.4, we include the link rewiring in our formalism and illustrate the effectivity of rewiring in absorbing the damage caused by churn upon the network. In section 6.5, we generalize the theory for the case where different peers join the network with individual/variable cutoff degrees. Section 6.6 validates the predictive power of the theoretical framework through accurate modeling of topological snapshot of Gnutella networks. The important findings which may be useful to the network engineers for improving the p2p services is summarized in section 6.7 after which we conclude the chapter. However, in order to develop the analytical framework, we build simple models of bootstrapping, rewiring and churn which is described next in section 6.2.

6.2 Modeling bootstrapping and other node/link dynamics

In this section, we model the three types node dynamics - bootstrapping, churn and rewiring. The incoming nodes join the network through bootstrapping protocols that is executed by different servent programs (limewire, gnucleus) [32, 82]. During bootstrapping, peer servent selects some online nodes guided by the 'good neighbor' criteria [93]. In order to attain the above objective, peers try (prefer) to join to high degree (bandwidth) nodes [62, 104]. We model bootstrapping protocols through node attachment rules where probability of attachment of the incoming peer to an online node is proportional to the degree of the online node. We realistically assume that bandwidth of a node is finite which restricts its maximum connectivity (*cutoff degree*). A node j, after reaching its cutoff degree $k_c(j)$, rejects any further connection requests from the incoming peers. Check Algorithm 6.1 for details.

Algorithm 6.1: Bootstrapping protocol executed by the joining node i

```
Node $i$ preferentially chooses $m'$ ($m' > m$) online nodes based on their current
degrees
while $m$ online nodes are not connected with $i$ do
    $j$ = select an online node among the chosen $m'$ nodes
    Node $i$ sends the connection request to $j$
    if degree(j)< $k_c(j)$ then
        Node $i$ connects with node $j$
    end
    else
        Node $j$ rejects the connection request
    end
end
```

Algorithm 6.2: Protocol executed by the departing node i

```
Node $i$ sends disconnection message to all the neighboring nodes
```

On the other hand, a fraction of nodes leave the network randomly either gracefully (notifying their neighbors, Algorithm 6.2) or abruptly. In such case, periodical pinging by online peers (peers which have not left the network) help themselves to keep updated about the status of their neighbors (Algorithm 6.3). In order to prevent network breakdown and to maintain quality of service, rewiring of the links take place by the online nodes at regular interval. They disconnect links from some of the connected peers and reconnect them with some good online peers (Algorithm 6.4). These three operations are now modeled with respect to a time step where at each timestep t, each of the three operations are performed with some probability

Algorithm 6.3: Topology maintenance protocol, periodically executed by all the online nodes in the network

```
foreach Node $i$ in the network do
    Send a ping message to all its neighbors and wait for the reply. If reply
    message is not received from a neighbor $j$ after a timeout period,
    disconnect the link $l_{ij}$
end
```

Algorithm 6.4: Rewiring protocol executed by the online node i

```
Node $i$ randomly selects a link $l_{ij}$ connected with node $j$
Send a disconnection request to node $j$ and disconnects the link $l_{ij}$
while Node $i$ has not established a new connection with node $j'$ do
    $j'$ = preferentially select an online node based upon the node degree
    Node $i$ sends the connection request to $j'$
    if $degree(j') < k_c(j')$ and $l_{ij'}$ does not exist then
        Node $i$ connects with node $j'$
    end
    else
        Node $j'$ rejects the connection request
    end
end
```

- With probability q, a new node joins the network following Algorithm 6.1.

- With probability r, a randomly selected node leaves the network due to peer churn. Peers in the network get updated about this activity either through Algorithm 6.2 or Algorithm 6.3.

- With probability w, a randomly chosen node in the network performs rewiring of a link following protocol specified in Algorithm 6.4.

This is important to note that $q + r + w$ is not necessarily 1. Since we deal with the growing network, the only restriction remains is $q > r$.

The analytical framework is developed next. Without loss of generality, here we assume that all the peers join the network with fixed cutoff degree k_c (sections 6.3, 6.4). However further in section 6.5, we generalize the formalism for the case where nodes may join with variable cutoff degrees.

6.3 Development of growth model in face of peer churn

In this section, we intend to compute degree distribution p_k (the fraction of k degree nodes in the networks) where nodes join the network through bootstrapping and leave through peer churn. These values of p_k can be computed by observing the shift in the number of k degree nodes to $k+1$ degree nodes as well as $k-1$ degree nodes to k degree nodes due to the attachment of a new node and removal of an existing node at timestep t. Let the fraction of nodes in the network having degree k at some timestep t be $p_{k,t}$, then the total number of k degree nodes before addition or removal of a node is $np_{k,t}$ (n is the total number of nodes at timestep t). After addition of the node with probability q and removal of the node with probability r, the total number of k degree nodes at timestep $t+1$ becomes $(n+q-r)p_{k,t+1}$. Hence, the change in the number of k degree nodes between the timesteps t and $t+1$ becomes

$$\Delta n_k = (n+q-r)p_{k,t+1} - np_{k,t} \tag{6.1}$$

It is assumed that asymptotically $p_{k,t+1} = p_{k,t} = p_k$ [11]. Hence the Eq. (6.1) becomes

$$\Delta n_k = (q - r)p_k \tag{6.2}$$

We formulate rate equations depicting these changes for some arbitrary degree k. By solving those rate equations, we calculate the degree distribution p_k of the entire network.

Joining of a node: In order to write rate equations [11], we need to know the probability A_k that an online node of degree k will receive a new link from the incoming peer. As stated in section 6.2, in this case the probability that an online node will receive an incoming link is proportional to the current degree k. So the probability that an online peer of degree k will receive a new link from the incoming peer is given by

$$\begin{aligned} A_k &= \frac{kp_k}{\sum_{k_1=0}^{k_c-1} k_1 p_{k_1}} = \frac{kp_k}{zf}, \quad k < k_c \\ &= 0, \qquad\qquad\qquad k \geq k_c \end{aligned} \tag{6.3}$$

where

$$f = \left(1 - \frac{k_c p_{k_c}}{z}\right) \tag{6.4}$$

is a parameter and $\sum_{k=0}^{k_c} kp_k = z$ is the average degree of the network. Here the denominator of Eq. (6.3) specifies the total number of edge tips (an edge has two tips) in the network excluding the nodes that have reached their cutoff degree k_c.

The addition of a new node of degree m at timestep $t+1$ changes the total number of k degree nodes in the network. This change can be formulated in the rate equation as the net change in the number of nodes of degree k in between timestep t and $t+1$. The mean number of nodes of degree k that gain an edge when a single new node of degree m joins the network at timestep $t+1$ is

$$\delta^{jo}_{k\to(k+1)} = m \times A_k = m\frac{kp_k}{zf} \tag{6.5}$$

On the other hand, the number of $k-1$ degree nodes that acquire a new edge each and become a node of degree k is

$$\delta^{jo}_{(k-1)\to(k)} = m\frac{(k-1)p_{k-1}}{zf} \tag{6.6}$$

Hence according to Eqs. (6.5) and (6.6), the net change in the number of k degree nodes due to joining of a new node

$$\delta^{jo}_{k} = \delta^{jo}_{(k-1)\to k} - \delta^{jo}_{k\to(k+1)} = m\left(\frac{(k-1)p_{k-1} - kp_k}{zf}\right) \tag{6.7}$$

Removal of a node: The removal of a node at timestep $t+1$ also changes the total number of k degree nodes in the network. Removal of a node affects the number of k degree nodes in three different ways; (a) removal of a k degree node itself (b) reduction in the number of k degree nodes due to the removal of a node that is neighbor of some k degree nodes: those nodes lose one link and move from degree k to $k-1$ (c) similarly increase in the number of k degree nodes due to removal of a node that is neighbor of $k+1$ degree nodes; a fraction of $k+1$ degree nodes move to k degree nodes.

Next we calculate the amount of decrease in the number of k degree nodes in the network due to the removal of r fraction of nodes where probability of removal of a k degree node is proportional to p_k. The probability of landing at a k degree node following a randomly chosen link can be designated as $\frac{kp_k}{\langle k\rangle}$ [127]. Subsequently, the average number of links of an arbitrarily chosen j degree node which are connected to the k degree nodes in the network can be expressed as $\frac{kp_k}{\langle k\rangle} \times j$. Hence, the average number of links in the network that lands at the k degrees nodes can be expressed as

$$A^{rm}_{k} = \sum_{j=0}^{k_c} p_j j \times \frac{kp_k}{\langle k\rangle} \tag{6.8}$$

Removal of a fraction of nodes in the network results in the removal of the links associated with them and subsequently the average number of k degree nodes that loses one link and become a node of degree $k-1$ is

$$\delta^{rm}_{k\to(k-1)} = \sum_{j=0}^{k_c} \frac{jp_j kp_k}{\langle k\rangle} = kp_k \tag{6.9}$$

Similarly due to removal of a node, a fraction of nodes having degree $k+1$ lose a link and move to the degree k. Hence according to Eq. (6.9), the change in the number of k degree nodes due to node removal

$$\begin{aligned} \delta_k^{rm} &= (-p_k + \delta_{(k+1)\to k}^{rm} - \delta_{k\to(k-1)}^{rm}) && (6.10) \\ &= (-p_k + (k+1)p_{k+1} - kp_k) = (k+1)[p_{k+1} - p_k] && (6.11) \end{aligned}$$

We now write the rate equations in order to formulate the change in the number of k degree nodes in the network due to the attachment of a new node of degree m with a probability q and removal of a node with probability r. Four pertinent degree ranges $k = 0$, $k = m$, $k \neq 0, m, k_c$ and $k = k_c$ are taken into consideration.

Rate equation for $0 < k < k_c$ such that $k \neq m$

According to the Eqs. (6.2), (6.7) and (6.10), the net change in the number of k degree nodes at timestep $t+1$ can be expressed as

$$\Delta n_k = q\delta_k^{jo} + r\delta_k^{rm} \tag{6.12}$$

Simplification of which results in

$$p_k = \frac{(\frac{qm(k-1)}{zf})p_{k-1} + (r(k+1))p_{k+1}}{q + rk + \frac{qmk}{zf}} \tag{6.13}$$

Rate equation for $k = m$

Beyond the normal increase in the m degree nodes by δ_m^{jo} and δ_m^{rm}, the entrance of the node itself with degree m adds an additional member in the m-degree node family. Hence similar to Eq. (6.12), the change in the number of m degree nodes at timestep $t+1$

$$\Delta n_m = q\left(1 + \delta_m^{jo}\right) + r\delta_m^{rm} \tag{6.14}$$

Subsequently,

$$p_m = \frac{q + (\frac{qm(m-1)}{zf})p_{m-1} + (r(m+1))p_{m+1}}{q + rm + \frac{qm^2}{zf}} \tag{6.15}$$

Rate equation for $k = k_c$

Since the network does not have any node of degree greater than k_c and nodes having degree k_c are not allowed to take any incoming links, nodes are only accumulated at degree $k = k_c$. However, a fraction of k_c degree nodes lose their links due to node removal and move to degree $k_c - 1$. Hence the rate equation

$$\Delta n_{k_c} = q\delta^{jo}_{(k_c-1)\to k_c} + r(-p_{k_c} - \delta^{rm}_{k_c\to(k_c-1)}) \tag{6.16}$$

Consequently, the corresponding recurrence becomes

$$p_{k_c} = \frac{(\frac{qm(k_c-1)}{zf})p_{k_c-1}}{q + rk_c} \tag{6.17}$$

Rate equation for $k = 0$

Nodes with degree $k = 0$ do not lose any link. However, nodes of degree 1 may lose one link due to node removal and move to degree 0. Hence, the rate equation becomes

$$\Delta n_0 = r(-p_0 + \delta^{rm}_{1\to0}) \tag{6.18}$$

Subsequently, we find

$$p_0 = \frac{r}{q}p_1 \tag{6.19}$$

The degree distribution p_k of the emerging network can be calculated by recursively solving the Eqs (6.13), (6.15), (6.17) and (6.19).

6.3.1 Special case: growth without peer churn

In this section, we consider a special case where the probability of peer churn becomes zero, that is $r = 0$. This fixes the minimum degree of the network to be m. The probability that an online peer of degree k will receive a new link from the incoming

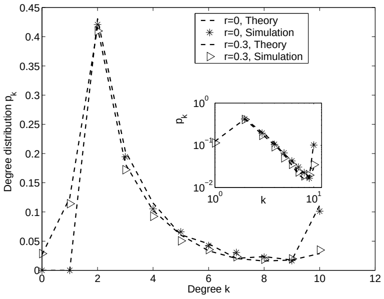


Figure 6.1: The degree distribution of the network emerged following the bootstrapping protocol with fixed cutoff degree $k_c = 10$ and $m = 2$ in face of peer churn r. Note, two peaks appearing at $k = 2$ and $k = 10$ respectively, spike at $k = 10$ illustrates the emergence of superpeer nodes. The dashed lines show the theoretical and the symbols show the simulation results. Inset shows the plot in log-log scale.

peer is given by

$$\begin{aligned} A_k &= \frac{kp_k}{\sum_{k_1=m}^{k_c-1} k_1 p_{k_1}} \quad k < k_c \\ &= 0 \qquad\qquad\qquad k \geq k_c \end{aligned} \tag{6.20}$$

Hence we get

$$A_k = \frac{kp_k}{\sum_{k_1=m}^{k_c-1} k_1 p_{k_1}} = \frac{kp_k}{2m - k_c p_{k_c}} = \frac{kp_k}{2mf} \tag{6.21}$$

where

$$f = (1 - \frac{k_c p_{k_c}}{2m}) \tag{6.22}$$

is a parameter and $\sum_{k=m}^{k_c} kp_k = 2m$ since there are m edges for each node added, and each edge, being now undirected, contributes two ends to the degrees of network nodes. Similar to the section 6.3, the rate equations are written for the following three regions $k = m$, $m < k < k_c$, $k = k_c$. According to Eq. (6.5), the mean number of nodes of degree k that gain an edge when a single new node of degree m joins the network at timestep $t+1$ is $\delta^{jo}_{k \to (k+1)} = m \times \frac{kp_k}{2mf} = \frac{kp_k}{2f}$, independent of m. On the other hand, $\delta^{jo}_{(k-1) \to k} = \frac{(k-1)p_{k-1}}{2f}$ number of nodes which were previously of degree $(k-1)$, acquire a new edge and become node of degree k.

Calculation of p_k for $k = m$

The joining of the node with degree m adds an additional member in the m-degree node family and removes on average $\frac{mp_m}{2f}$ nodes due to transition from degree m to $m+1$. Hence the net change in the number of nodes having degree $k = m$

$$(n+1)p_m - np_m = 1 - \frac{1}{2f}mp_m \tag{6.23}$$

Hence

$$p_m = \frac{2f}{2f+m} \tag{6.24}$$

Calculation of p_k for $m < k < k_c$

The net change in the number of nodes having degree k for $(m < k < k_c)$ due to the attachment of the new node

$$(n+1)p_{k,n+1} - np_{k,n} = \frac{1}{2f}(k-1)p_{k-1} - \frac{1}{2f}kp_k \tag{6.25}$$

Simplification of Eq. (6.25) results

$$\begin{aligned} p_k &= \frac{(k-1)}{(k+2f)}p_{k-1} & (6.26)\\ &= \frac{(k-1)(k-2)....m}{(k+2f)(k+2f-1)....(m+1+2f)}p_m & (6.27)\end{aligned}$$

Using (6.24), we get

$$\begin{aligned} p_k &= \frac{(k-1)(k-2)...m}{(k+2f)(k+2f-1)...(m+1+2f)} \times \frac{2f}{(2f+m)} \\ &= \frac{B(k,2f+1)}{B(2f+1,m)} \times \frac{2f}{2f+m} & (6.28)\end{aligned}$$

where $B(a,b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a+b)}$ is Legendre's beta function, which goes asymptotically as a^{-b} for large a and fixed b, and hence

$$p_k = \frac{k^{-(2f+1)}}{(2f+1)^{-m}} \times \frac{2f}{2f+m} \tag{6.29}$$

Calculation of p_k for $k = k_c$

The nodes having degree k_c are not allowed to accept connection from the incoming nodes. Hence,

$$(n+1)p_k - np_k \;=\; \frac{1}{2f}(k-1)p_{k-1} \tag{6.30}$$

which results

$$p_k = \frac{1}{2f}(k-1)p_{k-1} \tag{6.31}$$

Similarly using Eqs. (6.28) and (6.31) we get the following expression for $k = k_c$

$$\begin{aligned} p_{k_c} &= \frac{(k_c-1)(k_c-2)....m}{(k_c-1+2f)(k_c-2+2f)....(m+1+2f)}\frac{1}{(2f+m)} && (6.32)\\ &= \frac{B(k_c,2f)}{B(2f,m)} = \frac{k_c^{-2f}}{(2f)^{-m}} && (6.33)\end{aligned}$$

Using iterative substitution technique, we find the solution of f from the Eqs. (6.22) and (6.32). From the solution of f, we calculate p_k using Eqs. (6.24), (6.28), (6.32).

Fraction of superpeers in the network

Eq. (6.32) shows that the increase in the cutoff degree k_c reduces the percentage of superpeers in the network. This reduction of the amount of superpeers follows power law behavior with exponent $2f$.

Emergence of superpeer nodes

We are now in the position to theoretically understand the emergence of superpeer nodes. A closer look at the above equations points to two important observations. First, the fraction of nodes having degree k_c, p_{k_c}, is greater than p_{k_c-1}. From Eq.(6.31), we find

$$\frac{p_{k_c}}{p_{k_c-1}} = \frac{(k_c-1)}{2f} > 1 \tag{6.34}$$

Since $0 < f \leq 1$ and $k_c \gg 1$, the ratio $\frac{k_c-1}{2f} > 1$ subsequently $p_{k_c} > p_{k_c-1}$. The bootstrapping model gives $p_k = 0$ for $k > k_c$. Hence, we conclude the presence of a spike at degree k_c.

Secondly, we find for $m < k < k_c$, the probability continuously decreases. This can be understood from Eq. (6.26)

$$\frac{p_k}{p_{k-1}} = \frac{(k-1)}{(k+2f)} < 1 \tag{6.35}$$

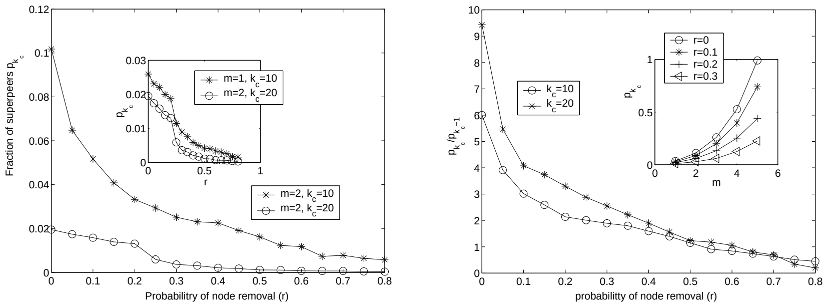
i.e. $p_k < p_{k-1}$. These two observations indicate the presence of two zones and direct to the emergence of high degree superpeer nodes at degree k_c. This is in line with the observations in previous section as well as in section 5.3.1 of Chapter 5.

6.3.2 Simulation results and inference derivation

We validate the theoretically obtained degree distribution by simulating the emergence of the network. The stochastic simulation set up is as follows. At each step, an incoming node gets connected to the network with probability q and some online node randomly gets removed from the network with probability r. We consider two different cases; in the first case, the removal probability r is set to 0. In second case, removal of node (with $r = 0.4$) also takes place in addition to the joining of nodes. In both cases, we fix the joining probability q at 1.0 which signifies that at each timestep, one node joins the network irrespective of removal and rewiring. The incoming node joins the networks with cutoff degree $k_c = 10$ and gets connected with two online nodes ($m = 2$) depending upon the current degree of that online nodes. The total number of nodes in the system is considered to be 5000 and we perform 500 individual realizations and plot the average degree distribution. We calculate the degree distribution for the typical case of $r = 0$ from section 6.3.1. Fig. 6.1 shows that the agreement between the theoretical and simulation results is exact which validates the correctness of the theoretical model. It is important to note the accumulation of superpeer nodes at degree 10.

6.3.3 Impact of peer churn

In the following, we investigate the influence of peer churn on the various topological properties of the emerging networks like largest component size, number of components and network diameter. We show that churn reduces the amount of superpeers in the networks and after a threshold value, churn destroys the bimodal structure of the emerging network. Churn also has a significant role in the disintegration of the largest component thus disrupting the communication among the peers.



(a) Sharp fall in the amount of superpeers (p_{k_c}) due to churn r for various k_c and m (inset).

(b) The increase in k_c improves critical churn threshold r_c (by increasing $\frac{p_{k_c}}{p_{k_c-1}}$ ratio). Inset shows that increase in the joining degree m increases the fraction of superpeers p_{k_c}.

Figure 6.2: The impact of peer churn (r) and joining degree (m) on the fraction of superpeers p_{k_c} and the ratio $\frac{p_{k_c}}{p_{k_c-1}}$.

Impact on superpeer nodes

Fig. 6.1 shows that in the absence of peer churn, a spike appears at around degree k_c which means the accumulation of superpeer nodes in the network. However, from the simulation results in Fig. 6.2(a), we find that the initial increase in r results in a sharp decrease in p_{k_c}. This happens due to two reasons; a) The presence of relatively high amount of superpeers leads to the initial spurt of their removal. b) random removal of nodes results in the disappearance of the links landing at the high degree superpeer nodes. The effect of these two dynamics reduces with the further increase in r as then p_{k_c} is already low. The Eq. (6.17) shows that p_{k_c} mainly depends upon the factor $\frac{k_c}{z} \approx \frac{k_c}{m}$ which is supported by the inset of Fig 6.2(a). Here two networks with identical $\frac{k_c}{m}$ ratio $(10, 1$ & $20, 2)$ have almost same amount of p_{k_c}.
As node removal probability r gets higher than the threshold r_c, the spike at $k = k_c$ disappears. The exact expression for r_c can be calculated as follows. The disappear-

ance of spike at k_c occurs when $p_{k_c} \leq p_{k_c-1}$, hence from the Eq. (6.17) we find

$$r_c \geq \frac{q}{k_c}\left(\frac{m(k_c-1)}{zf} - 1\right) \tag{6.36}$$

From the above expression, it becomes directly evident that increase in the cutoff degree k_c and joining degree m makes the spike at k_c more robust. In support of this fact, simulation results in Fig. 6.2(b) show that increase in k_c improves the $\frac{p_{k_c}}{p_{k_c-1}}$ ratio and subsequently the critical churn threshold r_c. Similarly, inset of Fig. 6.2(b) shows that increase in m sharply increases p_{k_c} and at $m = 5$, almost all the nodes in the network become superpeers. However, as the churn in the network increases, the p_{k_c} decreases.

Impact on the network component and diameter

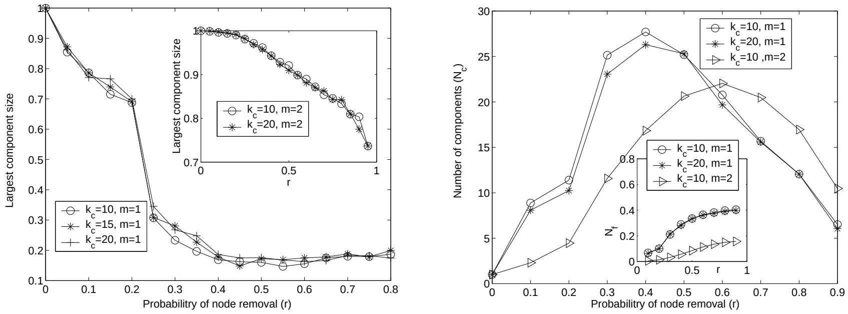
In Fig. 6.3, we analyze the effect of peer churn on the network component and the diameter. The largest connected component in the network plays a major role in the peer communication. For the sake of fairness, we define normalized number of components (N_f) and network diameter (N_d). N_f is derived by dividing the number of components N_c with the network size N whereas N_d as the ratio of shortest path length between two farthest nodes (d) and the expected diameter of the largest connected component LC (i.e. $\frac{d}{ln(LC)}$).

Initial network configuration: without churn

Fig. 6.3(a) shows that in the network without churn, there exists a single connected component where high degree superpeer nodes primarily drive the connectivity formation. The presence of a single connected component keeps the network diameter low (Fig. 6.4). The networks with higher cutoff degree (say $k_c = 20$) typically have low diameter due to the presence of high degree superpeer nodes.

Impact of small churn

The initial churn ($r = 0.1$) removes significant number of superpeer nodes (due to their considerable presence) which subsequently results in the *reduction of the largest component size* (Fig. 6.3(a)) and *increase in the number of components* in the network (Fig. 6.3(b)). As a result of churn, the amount of high degree superpeer nodes in the network reduces and in effect connectivity among the nodes within the largest con-



(a) The change in the largest component size with respect to the node removal. Inset shows that the largest connected component is more stable for $m = 2$.

(b) The change in the number of components (N_c) against peer churn r. In the inset, the number of components is measured as $N_f = \frac{N_c}{N}$ to show the behavior of N_c with respect to the network size (N) .

Figure 6.3: Fig. 6.3(a) and 6.3(b) show the impact of churn on the component formation in the network.

nected component weakens. This phenomenon can be conceptualized as the creation of 'holes' in the network as a result of churn.

Impact of heavy churn

$r > 0.2$ results in a *sharp fall in the largest connected component size* (Fig. 6.3(a)) and consequently the network gets disintegrated into a large number of *small disconnected components* (Fig. 6.3(b)). The dissolution of largest component happens due to the sudden percolation of 'holes' in the networks as a result of the removal of online nodes. It is interesting to note that the sudden percolation occurs independent of network size/cutoff degree, it only depends on average degree. The disintegration of the network increases the number of components in the network (Fig 6.3(b)) and even increases N_f (inset of Fig 6.3(b)). At $r > 0.5$, the number of components in the network decreases (Fig 6.3(b)) as many singleton components physically get removed, subsequently deaccelerating the increasing slope of N_f (inset of Fig. 6.3(b)).

Impact of m

The largest connected component of the network where nodes join with $m = 2$ exhibits

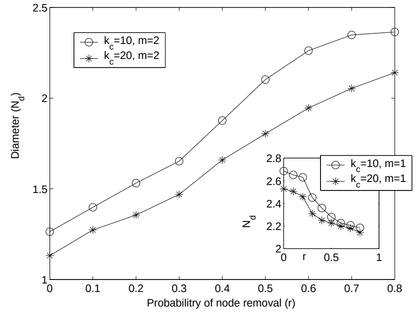


Figure 6.4: Change in diameter (N_d) due to the increase in r. The removal of key nodes increases N_d within the largest component. However, shrinking of the largest component results in the slight reduction in the diameter also (inset).

more stable behavior due to its high average degree (Inset of Fig. 6.3(a), Fig. 6.3(b)). However, the basic elegant properties of the network (like small network diameter) deteriorates. Fig. 6.4 shows that the network diameter increases for the largest stable component against churn due to the breakdown of the short length paths. Counter to this, we find a (slow) reduction in the diameter at $m = 1$ as the network itself begin to break down.

Summarization: In this section, we have developed a growth framework to analyze the emergence of superpeer networks against churn. We have observed that without churn, the network exhibits bimodal degree distribution where superpeer nodes appear at degree k_c as a spike and the amplitude of the spike reduces with k_c. However, further analysis have revealed that churn decreases the amount of superpeers in the network and after a threshold value, churn destroys the bimodality in the degree distribution. The impact of churn on the largest connected component and diameter depends upon the joining degree m. For instance, with $m = 1$, churn resulted in a sharp fall in the largest connected component size and consequently the network gets disintegrated into a large number of disconnected components. For $m > 1$, the network has showed stable behavior in terms of largest component size, however disappearance of shortest paths deteriorates the network diameter.

6.4 Development of growth model in face of peer churn and link rewiring

In this section, we include link rewiring in our growth model in addition to node joining and node removal. During rewiring, the disconnection of the old link and subsequent reconnection (Algorithm 6.4) do not change the total number of nodes and links in the network, but it significantly changes the topological structure and properties of the network. The assumption that all the nodes join the network with some fixed cutoff degree k_c is still valid here. We compute p_k by observing the shift in the number of k degree nodes to $k+1$ degree nodes as well as $k-1$ degree nodes to k degree nodes due to the attachment of a new node, removal of an existing node and rewiring of a link at time-step t. Similar to the previous case, asymptotically

$$\Delta n_k = (q-r)p_k \tag{6.37}$$

The addition, removal of nodes and rewiring of links may change the number of k degree nodes in the network in the following three ways.

Joining and removal of nodes: Similar to Eq. (6.7), the amount of increase in the k degree nodes due to the joining of a node may be expressed as

$$\delta_k^{jo} = m\left(\frac{(k-1)p_{k-1} - kp_k}{zf}\right) \tag{6.38}$$

Similarly, according to Eq. (6.10) the amount of increase in the number of k degree nodes due to node removal can be expressed as

$$\delta_k^{rm} = (-p_k + (k+1)p_{k+1} - kp_k) = (k+1)[p_{k+1} - p_k] \tag{6.39}$$

Rewiring of a link: Similar to addition and removal of a node, rewiring of a link also changes the total number of k degree nodes in the network. Let a randomly chosen node i be currently connected with the node j through a link l_{ij}. If the node i starts the relinking process, then it disconnects its connection with node j, preferentially chooses another node j' and establishes the new connection with node j' (if the node has not reached to its cutoff degree). Hence the rewiring leads to change in the number of node of degree k in two different ways; (a) link disconnection and (b) link reconnection.

(a) Due to the disconnection of the old link, a fraction of $k+1$ degree node loses one link and moves in to degree k and at the same time a fraction of k degree node loses one link and moves to degree $k-1$. We first calculate the fraction of k degree nodes that loses one link and moves to degree $k-1$. Probability of landing at one k degree node following a randomly chosen link is $\frac{kp_k}{z}$. We know that selecting a link connected to a randomly chosen node is equivalent to selecting a randomly chosen link in the network. Hence, mean number of k degree nodes in the network that loses one link due to the link disconnection and moves from degree k to $k-1$

$$\delta^{dis}_{k\to(k-1)} = \frac{kp_k}{z} \tag{6.40}$$

Hence the mean reduction in the k degree nodes due to the link disconnection process

$$\delta^{dis}_{k} = \delta^{dis}_{k\to(k-1)} - \delta^{dis}_{(k+1)\to k} = \frac{kp_k - (k+1)p_{k+1}}{z} \tag{6.41}$$

(b) On the other hand, the reconnection process of relinking also causes the change in the number of nodes of degree k as a k degree node (selected preferentially) accepts one new link (if its current degree is less than the cutoff degree k_c) from the node which initiates rewiring procedure. Similar to Eq. (6.5), the mean number of k degree nodes that accept a new link and move from degree k to $k+1$ becomes

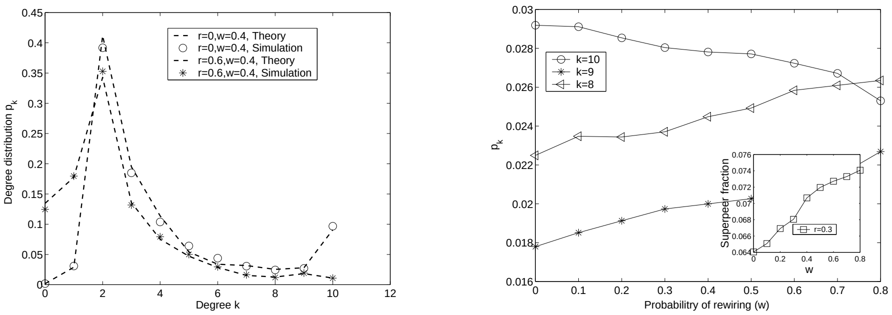
$$\delta^{recon}_{k\to(k+1)} = \frac{kp_k}{zf} \tag{6.42}$$

Hence the mean increase in the k degree nodes in the network due to preferential reconnection to the nodes of degree k

$$\delta^{recon}_{k} = \delta^{recon}_{(k-1)\to k} - \delta^{recon}_{k\to k+1} = \frac{(k-1)p_{k-1} - kp_k}{zf}$$

So the net increase in the number of k degree nodes in the network due to rewiring can be expressed as

$$\delta^{relink}_{k} = (\delta^{recon}_{k} - \delta^{dis}_{k}) \tag{6.43}$$



(a) Degree distribution of the emerging network in face of peer churn and link rewiring where $k_c = 10$ and $m = 2$.

(b) The impact of rewiring on the superpeer nodes of different degrees ($k = 8, 9, 10$). Inset shows that rewiring increases the fraction of superpeers in the network in face of churn.

Figure 6.5: Fig. 6.5(a) validates the theoretical results with simulation. Fig. 6.5(b) shows the impact of rewiring on the superpeer nodes.

We now write the rate equations in order to formulate the change in the number of k degree nodes in the network due to the attachment of a new node of degree m with a probability q, removal of a node with probability r and rewiring of links with probability w. Four pertinent degree ranges $k = 0$, $k = m$, $k \neq 0, m, k_c$ and $k = k_c$ are taken into consideration.

Rate equation for $0 < k < k_c$ such that $k \neq m$

The change in the k degree nodes in the network due to the joining (with probability q) or removal (with probability r) of nodes and rewiring of links (with probability w) can be expressed as

$$\Delta n_k = q\delta_k^{jo} + r\delta_k^{rm} + w\delta_k^{relink} \tag{6.44}$$

Subsequently the recurrence relation becomes

$$\psi_k p_k = \left(\frac{qm(k-1)}{zf} + \frac{w(k-1)}{zf}\right) p_{k-1} + \left(r(k+1) + \frac{w(k+1)}{z}\right) p_{k+1} \tag{6.45}$$

where

$$\psi_k = q(1 + \frac{mk}{zf}) + \frac{wk}{z}(1 + \frac{1}{f}) + rk \tag{6.46}$$

Rate equation for $k = m$

Following similar argument, the number of nodes having degree m, increases by one (the incoming node has degree m) in addition to the change in the m degree nodes by δ_m^{jo}, δ_m^{rm} and δ_m^{relink}. Hence similar to Eq. (6.12), the change in the number of m degree nodes in timestep $n + 1$

$$\Delta n_m = q(1 + \delta_m^{jo}) + r\delta_m^{rm} + w\delta_m^{relink} \tag{6.47}$$

Therefore we find

$$\psi_m p_m = q + \left(\frac{qm(m-1)}{zf} + \frac{w(m-1)}{zf}\right) p_{m-1} + \left(r(m+1) + \frac{w(m+1)}{z}\right) p_{m+1} \tag{6.48}$$

where $\psi_m = \psi_k$ at $k = m$.

Rate equation for $k = k_c$

Since the nodes having degree k_c are not allowed to take any incoming links, nodes are only accumulated at degree $k = k_c$. Hence

$$\Delta n_{k_c} = q\delta_{(k_c-1)\to k_c}^{jo} + r(-p_{k_c} - \delta_{k_c\to(k_c-1)}^{rm}) + w(\delta_{(k_c-1)\to k_c}^{recon} - \delta_{k_c\to(k_c-1)}^{dis}) \tag{6.49}$$

Therefore, the corresponding recurrence becomes

$$p_{k_c} = \frac{(\frac{qm(k_c-1)}{zf} + \frac{w(k_c-1)}{zf})p_{k_c-1}}{q + rk_c + \frac{wk_c}{z}} \tag{6.50}$$

Rate equation for $k = 0$

Nodes having degree $k = 0$ do not lose any link. Hence

$$\Delta n_0 = r(-p_0 + \delta_{1\to 0}^{rm}) + w\delta_{1\to 0}^{dis} \tag{6.51}$$

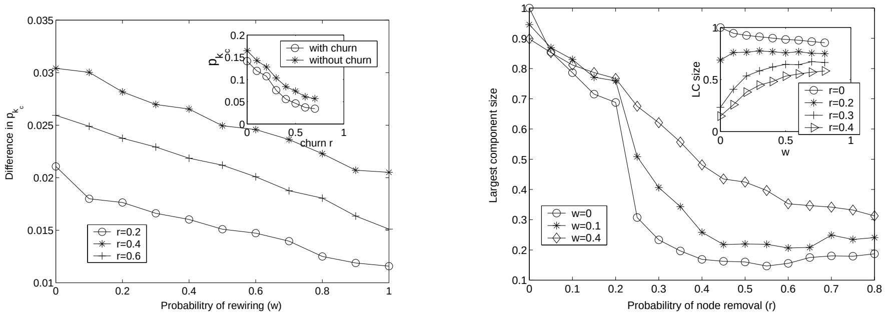
Subsequently

$$p_0 = \frac{\left(r + \frac{w}{z}\right)}{q} p_1 \tag{6.52}$$

We recursively use Eqs. (6.45), (6.48), (6.50) and (6.52) to compute the degree distribution p_k of the emerging networks.

6.4.1 Simulation results and inference derivation

We validate the theoretically obtained degree distribution by simulating the emergence of the network. During simulation, we assume that an incoming node gets connected with probability $q = 1$ to the network (with $k_c = 10$, $m = 2$), some random node gets removed from the network (with probability r) and link is rewired with probability w. We consider two different cases; in the first case, the removal probability r is set to 0. In second case, removal of node (with $r = 0.4$) also takes place in addition to the joining and rewiring of nodes. In both cases, we fix the joining probability q at 1.0 which signifies that at each timestep, one node joins the network irrespective of removal and rewiring. The total number of nodes in the system is considered to be 5000 and we perform 500 individual realizations and plot the average degree distribution. Fig. 6.5(a) shows that the agreement between the theoretical and simulation results is exact which validates the correctness of the theoretical model (dashed lines show the theoretical results whereas symbols depict the simulation results). Instead of having a single sharp degree (say 10) as in 'no churn, no rewiring' scenario, the superpeer region tapers off a bit and are distributed within a small value around the initial peak (say $8, 9, 10$). From the Fig. 6.5(a), the impact of rewiring on the superpeer nodes is not directly evident, Fig. 6.5(b) discusses that. In Fig 6.5(b), we show the impact of churn and rewiring on nodes around the peak, specifically we consider p_8, p_9 and p_{10}. The nodes of degree 10 lose links from churn and disconnection process of rewiring, hence its fraction reduces with w. However, nodes with degree 8 and 9 get benefited from the reconnection process of rewiring (due to their high degree) and p_8 and p_9 increases with w. In the inset, we show that rewiring continuously increases the superpeer fraction, mainly contributed from the nodes with degree 8 and 9. However, question remains whether rewiring is fully able to absorb the effect of churn in terms of the amount of superpeers still present.



(a) Rewiring to some extent absorbs the effect of churn on p_{k_c}. Inset shows that fraction of superpeers are less in network A (with churn) than network B (without churn).

(b) The change in the largest component size with respect to the node removal for different rewiring probabilities (w). Inset shows the change in the largest component with respect to w.

Figure 6.6: The impact of churn and rewiring on the fraction of superpeers in the network, largest component size and network diameter is shown.

Rewiring and amount of superpeers

In order to understand the impact of rewiring, we generate two networks A and B and compare the amount of superpeers present in them. Churn is simulated while generating network A, that is, at each step along with node joining ($m = 2$), there is a finite probability (r) of node removal. On the other hand, network B is generated only by the joining of incoming nodes so that the total number of nodes as well as the average degree of B becomes same as A. Inset of Fig 6.6(a) shows that fraction of superpeers in the network A (with churn) is always less than that of network B (without churn) and the difference remains almost constant. Next we perform rewiring on network A with probability w and subsequently record the change in the superpeer fraction. In Fig. 6.6(a), we show the difference between the fraction of superpeers in network B and the network A after rewiring. The results show that with the increase in rewiring probability w, the difference in the superpeer fraction between the networks A and B reduces, but it never reaches zero. Hence, we conclude that rewiring can absorb the effect of churn to some extent, however it fails to completely nullify the effect of churn.

Impact on the component formation and diameter

In section 6.3, Fig 6.3 showed that peer churn reduces the size of largest connected component and disintegrates the network into small components. Fig 6.6(b) shows the utility of rewiring in healing the largest connected component from churn whereby rewiring ensures that connectivity is maintained among the nodes of the network. In Fig 6.6(b), we show the impact in the largest connected component size against churn (r) for various rewiring probabilities (w).

Churn without rewiring: For $w = 0$, we observe a sharp fall in the largest connected component size (Fig. 6.6(b)) as nodes leave the network and disintegrates the network into smaller components. It is important to note that a few of these newly created components (apart from the largest connected component) are of moderate size, however rest of them are of very small size (mostly a singleton node).

Moderate rewiring gives benefit: In presence of proper rewiring, p2p network shows graceful degradation in face of churn. For example, at $r > 0.25$, nodes of the individual 'moderate size' components get connected by rewiring to form a larger connected component (Fig 6.6(b)). Hence the rewiring phenomenon neutralizes the effect of churn and saves the network from possible disintegration. However, the network diameter increases considerably. This is because the rewiring of existing links forms 'bridging links' between the 'moderate size' components in the network and through this, it merges the 'moderate sized components' into a larger connected component. Fig 6.6(b) shows that the increase in the rewiring probability gradually reduces the effect of churn on the largest component size and subsequently reduces the rate of reduction of the largest component size.

Heavy rewiring is not cost effective, sometimes detrimental: Inset of Fig. 6.6(b) indicates the existence of some crossover point such that if the churn rate is lower than some threshold value ($r = 0.073$), the rewiring of links may be detrimental. In this case, some of the existing nodes leave the largest connected component due to the disconnection of the links, which reduces the component size. On the other hand, it is important to note that above that specific churn rate, rewiring becomes helpful where rewiring integrates the disconnected components again through the reconnection procedure. However after some threshold level, the impact of rewiring saturates and further increase in w does not improve the network connectivity.

6.5 Formalism for variable cutoff degrees with peer churn and rewiring

Similar to section 5.4 of Chapter 5, we extend our formalism for the case, where nodes may join the network with individual/variable cutoff degrees. We assume that the probabilities that a node j joins the network with cutoff degree $k_c(j)$ is $q_{k_c(j)}$. Let every node necessarily have cutoff degree between a specified minimum and maximum, $k_c(min)$ and $k_c(max)$ respectively. Similar to Eq. 6.3, the probability that an online node of degree k receives a new link from the incoming peer or from another online peer (due to rewiring)

$$\begin{aligned}\widehat{A_k} &= \frac{kp_k}{\sum_{k=0}^{k_c(min)-1} kp_k + \sum_{k=k_c(min)}^{k_c(max)} kp_k S_k} \\ &= \frac{kp_k}{(z - \sum_{k=k_c(min)}^{k_c(max)} kp_k(1-S_k))} = \frac{kp_k}{zf_g} \end{aligned} \tag{6.53}$$

where

$$f_g = 1 - \frac{\sum_{k=k_c(min)}^{k_c(max)} kp_k(1-S_k)}{z} \tag{6.54}$$

implies the fraction of nodes in the network capable of accepting new links from the incoming peer and $z = \sum_{k=0}^{k_c} kp_k$ is the average degree of the network. Here S_k is the fraction of k degree nodes whose cutoff degree is greater than k and hence are still capable of taking incoming connections. We calculate the exact expression for S_k later in this section.

Similar to the fixed cutoff, we formulate the rate equations to characterize joining of an incoming node of degree m. Based on the behavior of S_k, the formulation of rate equation and subsequently the computation of degree distribution need to be done in two parts; nodes with degree $0 \leq k < k_c(min)$ in part A and nodes with degree $k_c(min) \leq k \leq k_c(max)$ in part B.

Part A : Dynamics analysis for $0 \leq k < k_c(min)$

In this case, none of the nodes has reached its cutoff degree. Hence S_k trivially becomes 1 and the rate equations for $0 \leq k < k_c(min)$ are similar to the Eqs. (6.12), (6.14), (6.16) and (6.18). Therefore, using these equations we calculate p_k.

Part B : Dynamics analysis for $k_c(min) \leq k \leq k_c(max)$

An important difference between part B and part A is that, at each k ($k_c(min) \leq k \leq k_c(max)$), a fraction of nodes reach their cutoff degree and stop taking further links from the incoming nodes. So the calculation of S_k becomes nontrivial and their values play a major role in formulating the rate equations. We start our analysis with the nodes having smallest cutoff degree $k = k_c(min)$.

Calculation for $k = k_c(min)$

We defined earlier that S_k is the fraction of nodes having degree $k = k_c(min)$ that have not reached their cutoff and still capable of taking incoming links. Hence similar to Eq. (6.5), on average $\widehat{\delta}^{jo}_{k\to(k+1)} = m \times \frac{kp_k}{zf_g} S_k$ number of nodes can move from degree $k_c(min)$ to $k_c(min) + 1$ because of addition of a new node with probability q and hence leave the $k_c(min)$ set. On the other hand, similar to Eq. (6.6), the mean number of nodes with degree $k-1$ that accepts new link and moves to degree k becomes $\widehat{\delta}^{jo}_{(k-1)\to k} = m \times \frac{(k-1)p_{k-1}}{zf_g}$. Hence in variable cutoff, the net change in the number of k degree nodes due to node joining

$$\begin{aligned} \widehat{\delta}^{jo}_k &= \widehat{\delta}^{jo}_{(k-1)\to k} - \widehat{\delta}^{jo}_{k\to(k+1)} \\ &= m\left(\frac{((k-1)p_{k-1} - kp_kS_k)}{zf_g}\right) \end{aligned} \tag{6.55}$$

The change in number of nodes having degree k due to the removal of a node can be calculated from Eq. (6.10)

$$\widehat{\delta}^{rm}_k = (-p_k + (k+1)p_{k+1} - kp_k) \tag{6.56}$$

In addition to that, the increase in the k degree nodes due to rewiring performed by a randomly selected node can be calculated from Eq. (6.43). However, it is important to note that the fraction of nodes of degree k that are able to accept new links due to reconnection process of rewiring can be expressed as $\frac{kp_k}{zf_g} S_k$. Hence, we appropriately change the reconnection expression of Eq. 6.43 as follows

$$\widehat{\delta}_k^{recon} = \left(\frac{(k-1)p_{k-1} - kp_kS_k}{zf_g}\right) \tag{6.57}$$

However, the mean reduction in the k degree nodes due to link disconnection process $\widehat{\delta}_k^{dis}$ remains same as Eq. (6.41). Subsequently the increase in the k degree node due to rewiring process can be expressed as

$$\widehat{\delta}_k^{relink} = (\widehat{\delta}_k^{recon} - \widehat{\delta}_k^{dis}) \tag{6.58}$$

$$= \left(\frac{(k-1)p_{k-1} - kp_kS_k}{zf_g}\right) - \left(\frac{kp_k - (k+1)p_{k+1}}{z}\right) \tag{6.59}$$

The net change in the number of nodes having degree k (for $k = k_c(min)$) due to the joining of a new node with probability q, removal of a node with probability r and rewiring of a link with probability w becomes

$$\Delta n_k = q\widehat{\delta}_k^{jo} + r\widehat{\delta}_k^{rm} + w\widehat{\delta}_k^{relink} \tag{6.60}$$

Calculation of S_k for $k = k_c(min)$

The joining of an incoming node (with probability q) and reconnection process of rewiring operation performed by the randomly chosen node (with probability w) result in the gain of the new links for some of the nodes in the networks. The mean number of nodes of degree $(k-1)$ that acquire the new links and move from degree $k-1$ to degree k due to joining and rewiring can be expressed as

$$q\widehat{\delta}_{(k-1)\to k}^{jo} + w\widehat{\delta}_{(k-1)\to k}^{recon} = (qm + w)\frac{(k-1)p_{k-1}}{zf_g} \tag{6.61}$$

Since q_k is the probability that a node joins the network with cutoff degree $k = k_c(min)$, the number of nodes that move from degree $k-1$ to k and also reach their cutoff degree $k = k_c(min)$ becomes

$$\widehat{\delta}_{(k-1)\to k}^{k_c(min)} = (q\widehat{\delta}_{(k-1)\to k}^{jo} + w\widehat{\delta}_{(k-1)\to k}^{recon}) \times \frac{q_k}{\sum_{k'=k}^{k_c(max)} q_{k'}} \tag{6.62}$$

The removal of node (with probability r) and disconnection process of rewiring (with probability w) results the movement of a fraction of $k+1$ degree nodes to $k = k_c(min)$ degree nodes. However, all these nodes have cutoff degree greater than $k_c(min)$, hence do not contribute in Eq. (6.62). As the fraction of k degree nodes in the network is p_k, then the fraction of nodes reaching the cutoff degree k after a particular timestep can be normalized as

$$1 - S_k = \frac{\frac{(qm+w)(k-1)p_{k-1}}{zf_g} q_k^*}{p_k} \Rightarrow S_k = 1 - \frac{(qm+w)(k-1)p_{k-1}q_k^*}{zf_g p_k} \tag{6.63}$$

where $q_k^* = \frac{q_k}{\sum_{k'=k}^{k_c(max)} q_{k'}}$. Substituting the value of S_k in Eq. (6.60) and rearranging p_k, we get

$$\begin{aligned} \widehat{\psi}_k p_k \;&=\; (qm+w)\frac{(k-1)}{zf_g}\left(1 + \frac{k(qm+w)q_k^*}{zf_g}\right)p_{k-1} \\ &\quad + \left(r(k+1) + \frac{w(k+1)}{z}\right)p_{k+1} \end{aligned} \tag{6.64}$$

where

$$\widehat{\psi}_k = (q + \frac{qmk}{zf_g} + rk + \frac{wk}{zf_g} + \frac{wk}{z}) \tag{6.65}$$

Calculation for $k = k_c(min) + 1$

This case differs from the previous ($k = k_c(min)$) in one aspect - unlike previous case, only $S_{k_c(min)}$ (i.e. S_{k-1}) fraction of $(k-1)$ degree nodes can accept incoming links (due to joining of the new node and rewiring operation performed by the existing node) and change their degree to k. However, similar to $k = k_c(min)$, S_k fraction of k degree nodes accept the new link and move to degree $k+1$. Hence, similar to Eqs. (6.55) and (6.59), the increase in the k degree nodes due to node joining and link rewiring can be expressed as

$$\widehat{\delta}_k^{jo} = \left(\frac{((k-1)p_{k-1}S_{k-1} - kp_kS_k)}{zf_g}\right) \tag{6.66}$$

and

$$\widehat{\delta}_k^{relink} = \left(\frac{((k-1)p_{k-1}S_{k-1} - kp_kS_k)}{zf_g}\right) - \left(\frac{kp_k - (k+1)p_{k+1}}{z}\right) \tag{6.67}$$

respectively. Subsequently following Eq. (6.60), we write the rate equation to compute the net change in the number of nodes having degree k due to the joining of a new node with probability q, removal a node with probability r and rewiring of a link with probability w.

Calculation of S_k for $k = k_c(min) + 1$

Similar to $k = k_c(min)$, the mean number of $(k-1)$ degree nodes that acquire the new links and move from the degree $(k-1)$ to degree k is

$$q\widehat{\delta}^{jo}_{(k-1)\to k} + w\widehat{\delta}^{recon}_{(k-1)\to k} = (qm+w)\frac{(k-1)p_{k-1}}{zf_g}S_{k-1} \tag{6.68}$$

Since q_k is the probability that a node joins the network with cutoff degree $k = k_c(min)+1$, the number of nodes that reaches the cutoff degree $k = k_c(min)+1$ after acquiring the new link may be expressed as

$$\widehat{\delta}^{k_c(min)+1}_{(k-1)\to k} = (q\widehat{\delta}^{jo}_{(k-1)\to k} + w\widehat{\delta}^{recon}_{(k-1)\to k}) \times q^*_k \tag{6.69}$$

With proper normalization, we find that the fraction of nodes that have not reached their cutoff degree $k = k_c(min) + 1$ and capable of taking incoming link

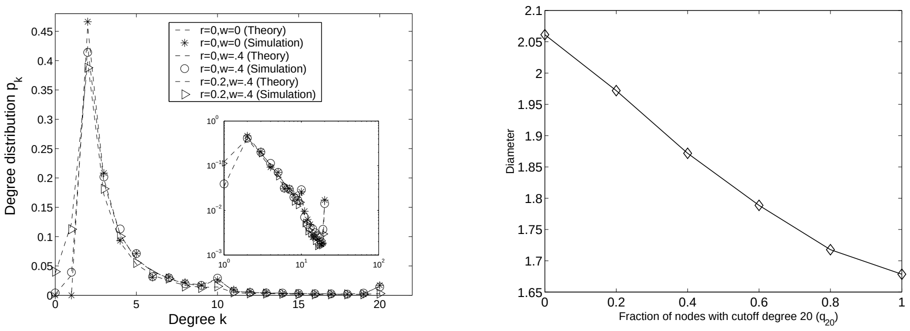
$$S_{k=k_c(min)+1} = 1 - \frac{(qm+w)\frac{(k-1)p_{k-1}}{zf_g}S_{k-1}q^*_k}{p_k} \tag{6.70}$$

Substituting the values of S_k, S_{k-1} in Eqs. (6.67), (6.66) and following Eq. (6.60), we get

$$\begin{aligned}\widehat{\psi}_k p_k \quad &= \quad (qm+w)\frac{(k-1)}{zf_g}\left(1+\frac{k(qm+w)q^*_k}{zf_g}\right)\left(p_{k-1} - \frac{(qm+w)(k-2)q^*_{k-1}}{zf_g}p_{k-2}\right) \\ &\quad + \left((k+1)(r+\frac{w}{z})\right)p_{k+1}\end{aligned} \tag{6.71}$$

Generalization : Continuing the calculations for $k_c(min) < k \leq k_c(max)$, we obtain the generalized equation

$$\begin{aligned}\widehat{\psi}_k p_k &= X(k-1)(1+Xkq^*)\left(p_{k-1} + \sum_{j=1}^{k-k_c(min)}(-X)^j\prod_{t=1}^{j}(k-t-1)q^*_{k-t}p_{k-t-1}\right) \\ &+ \left((k+1)(r+\frac{w}{z})\right)p_{k+1}\end{aligned} \tag{6.72}$$



(a) The variable cutoff model where 30% of peers join with cutoff degree 10 and 70% nodes join with cutoff degree 20.

(b) Increase in the fraction of nodes with cutoff degree 20 reduces the network diameter.

Figure 6.7: The degree distribution of the emerging network with and without peer churn (r) and rewiring (w) for variable cutoff degrees (Inset shows in log-log scale). Fig 6.7(b) shows the change in diameter where nodes joins with two cutoff degrees 10 and 20. The churn rate and rewiring probability are set to $r = 0.5$ and $w = 0$ respectively.

where

$$X = \frac{qm + w}{zf_g} \tag{6.73}$$

is a parameter which depends upon the node joining and link rewiring probability. The degree distribution of the network p_k can be calculated following Eqs (6.13), (6.15), (6.17) and (6.19) for $k < k_c(min)$, Eq. (6.64) for $k = k_c(min)$ and finally Eq. (6.72) for $k_c(max) \geq k > k_c(min)$.

Simulation results and inference derivation

In order to validate our theoretical framework, we simulate the bootstrapping protocol where nodes join with variable cutoff degrees. In our simulation, nodes can have 2 different cutoff degrees 10 and 20. We assume that the 30% of nodes join (say) through dial up lines with cutoff degrees 10. Rest 70% of nodes join (say) through

ISDN connection with cutoff degree 20. At the time of joining, each node establishes connections with 2 online nodes in the network i.e. $m = 2$. Similar to the previous cases, an incoming node gets connected with probability q to the network, a random node may get removed from the network with probability r and link is rewired with probability w. We consider two different cases; in the first case, the removal probability r and rewiring probability w is set to 0. In second case, removal of node (with $r = 0.2$) and rewiring of links (with $w = 0.4$) also takes place in addition to the joining of nodes. In both cases, we fix the joining probability q at 1.0 which signifies that at each timestep, one node joins the network irrespective of removal and rewiring. The total number of nodes in the system is 5000 and we perform 500 realizations. Fig 6.7(a) shows that the agreement between the theoretical model and simulation is exact.

Impact of cutoff degrees and their proportion: We investigate the impact of cutoff degrees and their individual fractions on the network topology. First we focus on the largest component size and next on the diameter. Fig. 6.3(a) shows that the change in the cutoff degree do not significantly change the behavior of largest connected component in the network. Hence we conclude that cutoff degrees and their individual fraction does not have much impact on component size. However, Fig 6.4 indicates that high cutoff degree k_c reduces the network diameter. In order to understand the role of individual cutoff fraction q_{k_c}, we perform a simulation where q_{20} fraction of nodes join with cutoff degree 20, and rest of the nodes join with cutoff degree 10. We set the churn rate to $r = 0.5$ and assume that rewiring is absent. In Fig 6.7(b), we show that for some given churn ($r = 0.5$) and rewiring probability ($w = 0$), the increase in the fraction of joining nodes with cutoff degree 20 reduces the network diameter. The high degree nodes ($k > 10$) in the network play a crucial role in reducing the network diameter.

6.6 Case study with Gnutella network

In order to illustrate the predictive power of the theoretical model, we chose to investigate the topological snapshot of Gnutella networks. As explained in Chapter 3,

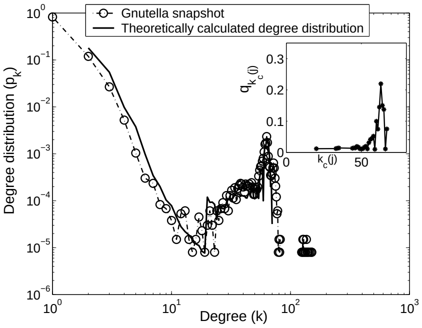


Figure 6.8: The figure illustrates the comparative study between the real world Gnutella networks [1] and our theoretical model. The inset shows the cutoff degree distribution $q_{k_c(j)}$ that provides excellent fit of our calculated degree distribution with the real network data.

we simulate Gnutella network following the snapshot obtained from the Multimedia & Internetworking Research Group, University of Oregon, USA [1](Fig. 6.8). In order to check whether the degree distribution of Gnutella can be explained through the developed framework, we theoretically compute the degree distribution of the emerging network. Since during connection initiation, most of the servents initially connect to multiple online peers [82], therefore we keep $m = 2$. The Gnutella network consistently grows as a net effect of the joining of the new nodes and peer churn. The rewiring of existing links also changes the topological structure of the networks. We describe the evolution of Gnutella network due to joining, removal of nodes and rewiring of links by the tuple (q, r, w). We keep $q = 1$ to signify that at each timestep, one new node joins the network. To obtain r and w, we fit the calculated degree distribution with Gnutella snapshot, obtaining an excellent overlap for $r = 0.474$ and $w = 0.249$. Similarly, the probability $q_{k_c(j)}$ that a node j joins with cutoff degree $k_c(j)$ is adjusted accordingly to fit the calculated degree distribution close to the Gnutella network (inset of Fig. 6.8). As can be seen from Fig. 6.8, our theoretical model can mimic the degree distribution of Gnutella network with reasonable accuracy. The results indicate that on average 47.4% of nodes in Gnutella leave the network due to peer churn. However, the network survives due to the significant amount of rewiring (24.9%) performed by the online nodes. This theoretical result is reinforced by the measurement study of [163] on the dynamics of Gnutella networks which also reports heavy churn.

6.7 Conclusion

In this chapter, we have extended the analytical framework developed in Chapter 5 to explain the evolution of superpeer networks in face of peer churn and link rewiring. Our results have shown that a small churn results in a sharp reduction in the superpeer fraction and after a threshold amount of churn, the bimodality of the degree distribution disappears. In addition to that, churn also results in a sharp fall in the largest connected component size and consequently the network gets disintegrated into a large number of disconnected components. Interestingly, the breakdown syndrome seems to be independent of the topological properties of the network. The servent program may be suitably designed to heal the network and maintain the QoS by performing proper rewiring operation. Rewiring helps in absorbing some of the damages caused by churn, however one should not do it too early nor overdo it; both may be detrimental. But expectation about the impact of rewiring should be reasonable as rewiring does not fully heal the damage created by churn. The best part of our framework lies in the excellent match it produces with respect to Gnutella network's degree distribution. We feel that this complete comprehensive framework will be used by design engineers to understand the impact of various parameters and accordingly design better, more robust and efficient peer-to-peer networks in the future.

This brings to the end of the contributory chapters. We summarize our contributions and conclude the thesis in the next chapter.

Chapter 7

Conclusion and Future work

In this chapter, we summarize the main contributions of the thesis and take a stock of our achievements vis–a–vis the objectives set up in the introduction of the thesis. We find that the objectives have been largely achieved. We also realize various shortcomings of our work and identify unfinished agenda which we put forward as future work.

7.1 Summary of our contributions

In this thesis, our contributions are two-fold (a) Building up comprehensive theoretical frameworks characterizing the stability and emergence of superpeer networks, and (b) Reporting nonintuitive observations which arise from the interplay of the underlying parameters. We can thus broadly categorize the contributions in terms of (a) developing models and (b) carrying out extensive analysis and drawing inferences upon the developed models.

7.1.1 Stability analysis

Modeling: We have proposed an analytical framework that can predict the extent of connectivity preserved among the nodes in face of churn and attacks (Chapters 3 and 4). We have modeled peer churn and attacks as the removal of nodes from the network. Peer churn is characterized by degree independent and degree dependent failures and attacks by deterministic and degree dependent attacks. We have shown that our framework is capable of predicting the degree distribution of the deformed topology after attack and also can take the degree-degree correlation present in the network under consideration. The validation of the theoretical results is done both by simulating random graphs and using real world Gnutella network.
Analysis: Rigorous analysis reveals the following interesting observations

1. Superpeer networks exhibit stable behavior against user churn, which is consistent with the various measurement studies [69, 146].

2. In deterministic attack, networks having low peer degree are very much vulnerable and removal of only a small fraction of superpeers causes the breakdown of the network. But as the peer degree increases, the stability of the network increases as well.

3. In degree dependent attack $f_k \sim k^{-\gamma}$, we have formulated a critical condition whose solution set provides the critical exponent $\gamma = \gamma_c$. The peers and superpeers required to be removed is dependent upon this critical exponent γ_c.

4. We have shown that, at some typical γ_c, degree dependent attack reduces to deterministic attack. Subsequently, any kind of node removal technique can be expressed as the degree dependent attack.

5. We have further analyzed the degree dependent attack and found that available information about the network topology makes the attack efficient by reducing the percolation threshold. However, beyond a threshold limit, this information does not help the attackers in a significant manner.

6. The simulations on the small sized network has also shown that classical theory

overestimates the percolation threshold, however, our equation provides a better approximation.

7.1.2 Network emergence

Modeling: We have investigated the reason behind the emergence of superpeer networks, where in the networks, incoming nodes join through servents, randomly leaves the network due to churn and restructure their neighborhood through rewiring of links. First, in Chapter 5, we have considered the emergence of the network only through node joining, further we included peer churn and link rewiring in our formalism in Chapter 6. In Chapter 5, we have modeled the bootstrapping protocol through node attachment rule where the probability of joining of an incoming peer to an online node is proportional to the resources like processing power, storage space etc as well as current degree of the online node. We have shown that the interplay of finiteness of bandwidth with node resource play a key role in the emergence of bimodal superpeer network. In Chapter 6, we have developed a more generalized growth framework where nodes may undergo various kinds of dynamics like bootstrapping, peer churn, link rewiring etc. In order to keep the calculation simple, in Chapter 6, we have characterized the 'goodness' of a node only by its current degree. The degree distribution of the emerging network calculated through the generalized growth framework has exhibited nice agreement with simulation results as well as real Gnutella snapshot.

Analysis: Rigorous analysis of the growth framework leads to some interesting observations

1. Increase in the resourceful nodes may increase the fraction of superpeers only upto a level, however presence of too many high resource nodes may be detrimental.

2. GWebCache is periodically populated by the online peers/superpeers nodes based on the specific servent implementation. Carefully modifying the bootstrapping protocol to sieve appropriate nodes from the GWebCache may im-

prove the p2p services by reducing search latency and enhancing the speed of file download etc.

3. Instead of joining the network with different bandwidth levels, using a few (or single) cutoff degrees is optimal for the emergence of high amount of superpeers in the network.

4. Small churn results in a sharp reduction in the superpeer fraction and after a threshold amount of churn, the bimodality of the degree distribution disappears.

5. Churn may deteriorate the performance by disintegrating the network in components. However, in presence of proper rewiring, superpeer network shows graceful degradation in face of churn; the nodes largely remain connected, but the diameter of the network increases. Rewiring acts as the 'bridging links' between the 'moderate size' components in the network.

6. If the churn rate is lower than some threshold, rewiring itself may be detrimental as disconnection of links removes smaller components from the network. On the other extreme, beyond a threshold level, the impact of rewiring saturates and further increase does not improve the network connectivity.

7. Finally, the comparative study of our growth framework with the real world Gnutella network has provided some estimation of the nature of the nodes of the network, the churn and rewiring rate etc.

7.2 Future directions

In this final section, we discuss few of the many possible directions that have been opened up by this thesis.

1. We have modeled churn as the removal of nodes (either randomly or based upon degree) along with the adjacent links. This churn model may be extended if we include the 'lifetime' or 'session time' of a peer in consideration. This will

make the churn analysis more sophisticated and unfold the impact of different factors on the network stability.

2. In Chapter 4, we have proposed a basic framework for calculating p'_k in correlated networks. This framework may be extended further to derive the critical condition as well as to calculate percolation threshold in correlated network. There are many interesting questions that need to be addressed for correlated network. For instance, (a) do all the networks of a given correlation coefficient exhibit same amount of stability? In that line, one may come up with a unified metric which may capture both degree correlation and stability in a single parameter. (b) We have shown that attack in correlated network may change the density (average degree) of the network. This may lead to some attack (node removal) strategies altering with the network density and subsequently affecting QoS.

3. In network emergence, we have modeled the joining of incoming nodes as the GWebCache based bootstrapping protocols. However, there are several other bootstrapping strategies like random address probing, locality aware bootstrapping etc that need to be investigated and modeled.

4. In Chapter 6, we have assumed that during churn, a node leaves the network along with its adjacent links. However, in some cases, churn of a node leads to the formation of new links across the neighboring peers to keep their degree constant. Modeling this churn dependent rewiring mechanisms in a growing network may be the future work.

5. In this thesis, we have used the snapshot of Gnutella network of September 2004 to validate our frameworks. This surely act as a first proof of concept. However, several recent topological snapshots of other popular p2p networks like edonkey, KaZaA, skype etc should also be used.

6. In Chapters 5 and 6, we have developed a formalism to explain the emergence of bimodal superpeer network due to joining, leaving of nodes and restructuring of links in the context of superpeer network. This formalism may be further used to analyze the growth of other social networks exhibiting similar kind of dynamics.

Bibliography

[1] Gnutella snapshot. http://mirage.cs.uoregon.edu/p2p/info.cgi.

[2] Gnutella webcache scan report - http://gcachescan.jonatkins.com/.

[3] Limewire. http://www.limewire.com.

[4] Napster: http://www.napster.com/.

[5] L. A. Adamic and B. A. Huberman. Power-law distribution of the world wide web. *Science*, 287:2115, 2000.

[6] L. A. Adamic, R. M. Lukose, A. R. Puniyani, and B. A. Huberman. Search in power-law networks. *Physical Review E*, 64(4):046135, Sep 2001.

[7] R. Albert and A. Barabási. Topology of evolving networks: Local events and universality. *Physical Review Letters*, 85(24):5234, December 2000.

[8] R. Albert, H. Jeong, and A.-L. Barabasi. The diameter of the world wide web. *Nature*, 401:130, 1999.

[9] R. Albert, H. Jhong, and A. L. Barabasi. Error and attack tolerance of complex networks. *Nature*, 406, 2000.

[10] N. Alon, I. Benjamini, and A. Stacey. Percolation on finite graphs and isoperimetric inequalities. *Annals of Probability*, 32(3):1727–1745, 2004.

[11] A. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, October 1999.

[12] E. Ben-Naim and P. L. Krapivsky. Addition-deletion networks. *Journal of Physics A: Mathematical and Theoretical*, 40:8607, 2007.

[13] N. Berger, C. Borgs, J. T. Chayes, and A. Saberi. On the spread of viruses on the internet. In *SODA '05: Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 301–310, Philadelphia, PA, USA, 2005.

[14] A. Beygelzimer, G. Grinstein, R. Linsker, and I. Rish. Improving network robustness by edge modification. *Physica A: Statistical Mechanics and its Applications*, 357:593–612, 2005.

[15] R. Bhagwan, S. Savage, and G. M. Voelker. Understanding availability. In *IPTPS 2003: Proceedings of the Second International Workshop on Peer-to-Peer Systems II, Berkeley, CA, USA, February 21-22*, pages 256–267, 2003.

[16] G. Bianconi. Emergence of weight-topology correlations in complex scale-free networks. *Europhysics Letters*, 71(6):1029–1035, 2005.

[17] G. Bianconi and A.-L. Barabasi. A bose-einstein condensation in complex networks. *Physical Review Letters*, 86:5632–5635, 2001.

[18] G. Bianconi and A. L. Barabasi. Competition and multiscaling in evolving networks. *Europhysics Letters*, 54(4):436–442, 2001.

[19] J. Billen, M. Wilson, A. Rabinovitch, and A. R. C. Baljon. Topological changes at the gel transition of a reversible polymeric network. *Europhysics Letters*, 87(68003), 2009.

[20] R. Bolla, R. Gaeta, A. Magnetto, M. Sciuto, and M. Sereno. A measurement study supporting p2p file-sharing community models. *Computer Networks*, 53(4):485–500, March 2009.

[21] D. S. Callaway, J. E. Hopcroft, J. M. Kleinberg, M. E. J. Newman, and S. H. Strogatz. Are randomly grown graphs really random? *Physical Review E*, 64(4):041902, Sep 2001.

[22] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts. Network robustness and fragility: Percolation on random graphs. *Physical Review E*, 85(21), 2000.

[23] M. Castro, P. Druschel, A. Ganesh, A. Rowstron, and D. S. Wallach. Secure routing for structured peer-to-peer overlay networks. *ACM SIGOPS Operating Systems Review*, 36(SI):299–314, 2002.

[24] M. Castro, P. Druschel, A.-M. Kermarrec, and A. Rowstron. One ring to rule them all: service discovery and binding in structured peer-to-peer overlay networks. In *EW 10: Proceedings of the 10th workshop on ACM SIGOPS European workshop*, pages 140–145, New York, NY, USA, 2002.

[25] Y. P. Chen, G. Paul, R. Cohen, S. Havlin, S. P. Borgatti, F. Liljeros, and H. E. Stanley. Percolation theory and fragmentation measures in social networks. *Physica A: Statistical Mechanics and its Applications*, 378(1), 2007.

[26] N. Christin, A. S. Weigend, and J. Chuang. Content availability, pollution and poisoning in file sharing peer-to-peer networks. In *Proceedings of the 6th ACM Conference on Electronic Commerce*. Vancouver, BC, Canada, June 5-8 2005.

[27] D. Clark. Face-to-face with peer-to-peer networking. *IEEE Computer*, 34(1):18–21, 2001.

[28] R. Cohen, K. Erez, D. Avraham, and S. Havlin. Resilience of the internet to random breakdown. *Physical Review Letters*, 85(21), 2000.

[29] R. Cohen, K. Erez, D. Avraham, and S. Havlin. Resilience of the internet under intentional attack. *Physical Review Letters*, 86(16), 2001.

[30] R. Cohen, S. Havlin, and D. ben Avraham. Efficient immunization strategies for computer networks and populations. *Physical Review Letters*, 91(24):247901, Dec 2003.

[31] M. Conrad and H.-J. Hof. A generic, self-organizing, and distributed bootstrap service for peer-to-peer networks. In *IWSOS 2007: Proceedings of the Second International Workshop Self-Organizing Systems*, pages 59–72, The Lake District, UK, 2007.

[32] M. Conrad and H.-J. Hof. A generic, self-organizing, and distributed bootstrap service for peer-to-peer networks. *Self-Organizing Systems*, pages 59–72, 2007.

[33] F. Cornelli, E. Damiani, S. D. C. di Vimercati, S. Paraboschi, and P. Samarati. Choosing reputable servents in a p2p network. In *WWW '02: Proceedings of the 11th International Conference on World Wide Web*, pages 376–386, New York, NY, USA, 2002.

[34] C. Cramer, K. Kutzner, and T. Fuhrmann. Bootstrapping locality-aware p2p networks. In *ICON 2004: Proceedings of the IEEE International Conference on Networks*, pages 357–361, 2004.

[35] P. Crucitti, V. Latora, and M. Marchiori. Model for cascading failures in complex networks. *Physical Review E*, 69(4):045104, Apr 2004.

[36] P. Crucitti, V. Latora, M. Marchiori, and A. Rapisarda. Efficiency of scale-free networks: error and attack tolerance. *Physica A: Statistical Mechanics and its Applications*, 320:622–642, March 2003.

[37] E. Damiani, S. Paraboschi, P. Samarati, and F. Violante. A reputation based approach for choosing reliable resources in peer to peer networks. In *ACM CCS 2002: Proceedings of the 9th ACM Conference on Computer and Communications Security*. Washington, DC, USA, November 18-22 2002.

[38] T. Das, S. Nandi, and N. Ganguly. Community based search on power law networks. In *COMSWARE 2008: Proceedings of the Third International Conference on COMmunication System softWAre and MiddlewaRE, January 5-10, 2008, Bangalore, India*, pages 279–282, 2008.

[39] D. J. de Solla Price. Networks of scientific papers. *Science*, 149:510–515, 1965.

[40] D. J. de Solla Price. A general theory of bibliometric and other cumulative advantage processes. *Journal of the American Society for Information Science*, 27:292–306, 1976.

[41] Q. Deng and H. Lv. Analyzing unstructured peer-to-peer search networks with qil. In *IEEE SCC '04: Proceedings of the IEEE International Conference on Services Computing*, pages 547–550, Washington, DC, USA, 2004.

[42] S. N. Dorogovtsev and J. F. F. Mendes. Evolution of networks with aging of sites. *Physical Review E*, 62(2):1842–1845, Aug 2000.

[43] S. N. Dorogovtsev and J. F. F. Mendes. Evolution of networks with aging of sites. *Physical Review E*, 62(2):1842–1845, 2000.

[44] S. N. Dorogovtsev and J. F. F. Mendes. Scaling behaviour of developing and decaying networks. *Europhysics Letters*, 52:33–39, 2000.

[45] S. N. Dorogovtsev, J. F. F. Mendes, and A. N. Samukhin. Structure of growing networks with preferential linking. *Physical Review Letters*, 85:4633–4636, 2000.

[46] J. R. Douceur. The sybil attack. In *IPTPS '01: Revised Papers from the First International Workshop on Peer-to-Peer Systems*, pages 251–260, London, UK, 2002.

[47] D. Dumitriu, E. Knightly, A. Kuzmanovic, I. Stoica, and W. Zwaenepoel. Denial-of-service resilience in peer-to-peer file sharing systems. *ACM SIGMETRICS Performance Evaluation Review*, 33(1):38–49, 2005.

[48] P. Erdos and A. Renyi. On random graphs I. *Publicationes Mathematicae, Debrecen*, 6, 1959.

[49] P. Erdos and A. Renyi. On the evolution of random graphs. *Publication of the Mathematical Institute of the Hungarian Academy of Sciences*, 5, 1960.

[50] G. Ergun and G. J. Rodgers. Growing random networks with fitness. *Physica A: Statistical Mechanics and its Applications*, 303:261272, 2002.

[51] A. Fabrikant, E. Koutsoupias, and C. H. Papadimitriou. Heuristically optimized trade-offs: a new paradigm for power laws in the internet. In *Proceedings of the 29th International Conference on Automata, Languages and Programming*, pages 110–122, 2002.

[52] T. Fenner, M. Levene, and G. Loizou. A stochastic evolutionary model exhibiting power-law behaviour with an exponential cutoff. *Physica A: Statistical Mechanics and its Applications*, 355(2-4):641–656, September 2005.

[53] T. Fenner, M. Levene, and G. Loizou. A model for collaboration networks giving rise to a power-law distribution with an exponential cutoff. *Social Networks*, 29(1):70–80, January 2007.

[54] L. Gallos, P. Argyrakis, A. Bunde, R. Cohen, and S. Havlin. Tolerance of scale-free networks : from friendly to intentional attack strategies. *Physica A: Statistical Mechanics and its Applications*, 344(3–4):504–509, 2004.

[55] L. K. Gallos, R. Cohen, P. Argyrakis, A. Bunde, and S. Havlin. Stability and topology of scale-free networks under attack and defense strategies. *Physical Review Letters*, 94(188701), 2005.

[56] L. K. Gallos, R. Cohen, P. Argyrakis, A. Bunde, and S. Havlin. Stability and topology of scale-free networks under attack and defense strategies. *Physical Review Letters*, 94(188701), 2005.

[57] L. K. Gallos, K. Lazaros, R. Cohen, F. Lijeros, P. Argyrakis, A. Bunde, and S. Havlin. Attack strategies on complex networks. In *International Conference on Computational Science - ICCS 2006*, volume 3993. UK, December 12-13 2006.

[58] N. Ganguly and A. Deutsch. Developing efficient search algorithms for p2p networks using proliferation and mutation. In *Proceedings of the International Conference on Artificial Immune Systems*. Catania, Italy, September 2004.

[59] P. Garbacki, D. H. J. Epema, and M. van Steen. A two-level semantic caching scheme for super-peer networks. In *WCW '05: Proceedings of the 10th International Workshop on Web Content Caching and Distribution*, pages 47–55, Washington, DC, USA, 2005.

[60] P. Garbacki, D. H. J. Epema, and M. van Steen. Optimizing peer relationships in a super-peer network. In *ICDCS '07: Proceedings of the 27th International Conference on Distributed Computing Systems*, page 31, Washington, DC, USA, 2007.

[61] C. GauthierDickey and C. Grothoff. Bootstrapping of peer-to-peer networks. In *Proceedings of DAS-P2P*, Turku, Finland, August 2008.

[62] C. GauthierDickey and C. Grothoff. Bootstrapping of peer-to-peer networks. In *DAS-P2P 2008: IEEE Proceedings of the International Workshop on Dependable and Sustainable Peer-to-Peer Systems*, Turku, Finland, August, 2008.

[63] X. Geng and Q. Li. Random models of scale-free networks. *Physica A: Statistical Mechanics and its Applications*, 356:554–562, Oct. 2005.

[64] G. Ghoshal and M. E. J. Newman. Growing distributed networks with arbitrary degree distributions. *European Physical Journal B*, 59:75, 2007.

[65] Gnutella. http://www.gnutella.com.

[66] P. B. Godfrey, S. Shenker, and I. Stoica. Minimizing churn in distributed systems. *SIGCOMM Computer Communication Review*, 36(4):147–158, 2006.

[67] A. V. Goltsev, S. N. Dorogovtsev, and J. F. F. Mendes. Percolation on correlated networks. *Physical Review E*, 78(5):051105, Nov 2008.

[68] H. Guclu, D. Kumari, and M. Yuksel. Ad hoc limited scale-free models for unstructured peer-to-peer networks. In *P2P '08: Proceedings of the Eighth International Conference on Peer-to-Peer Computing*, pages 160–169, Washington, DC, USA, 2008.

[69] S. Guha, N. Daswani, and R. Jain. An experimental study of the skype peer-to-peer voip system. *IPTPS '06: Proceedings of The 5th International Workshop on Peer-to-Peer Systems*, pages 1–6, 2006.

[70] J.-L. Guillaume, M. Latapy, and C. Magnien. Comparison of failures and attacks on random and scale-free networks. In *OPODIS 2004: 8th International Conference on Principles of Distributed Systems, Grenoble, France*, pages 186–196, 2004.

[71] K. Hildrum and J. Kubiatowicz. Asymptotically efficient approaches to fault-tolerance in peer-to-peer. In *DISC 2003: Proceedings of the 17th International Symposium On Distributed Computing*, pages 321–336, Sorrento, Italy, 2003.

[72] P. Holme. Efficient local strategies for vaccination and network attack. *Europhysics Letters*, 68(6), 2004.

[73] P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han. Attack vulnerability of complex networks. *Physical Review E*, 65(5):056109, May 2002.

[74] X. Huang, Y. Li, R. Yang, and F. Ma. Enhancing attack survivability of gnutella-like p2p networks by targeted immunization scheme. In *PDCAT 2005: Sixth International Conference on Parallel and Distributed Computing, Applications and Technologies, 5-8 December 2005, Dalian, China*, pages 503–506, 2005.

[75] X. Huang, F. Ma, and Y. Li. Attack vulnerability of peer-to-peer networks and cost-effective immunization. In *Parallel and Distributed Processing and Applications - ISPA 2005 Workshops*, volume 3759 of *LNCS*, pages 45–53, 2005.

[76] F. S. Inc. Superpeer architectures for distributed computing. In *White Paper, http://www.fiorano.com/whitepapers/ superpeer.pdf*, 2002.

[77] M. Jelasity, A. Montresor, and O. Babaoglu. The bootstrapping service. In *ICDCSW '06: Proceedings of the 26th IEEE International ConferenceWorkshops on Distributed Computing Systems*, page 11, Lisboa, Portugal, 2006.

[78] M. Jelasity, A. Montresor, and O. Babaoglu. T-man: Gossip-based fast overlay topology construction. *Computer Networks*, 53(13):2321–2339, August 2009.

[79] G. P. Jesi, A. Montresor, and O. Babaoglu. Proximity-aware superpeer overlay topologies. In *Proceedings of the 2nd IEEE International Workshop on SelfMan, LNCS 3996*, pages 43–57, 2006.

[80] M. A. Jovanovic, F. S. Annexstein, and K. A. Berman. Modeling peer-topeer network topologies through small-world models and power laws. In *IX Telecommunications Forum*, 2001.

[81] G. Kan. Gnutella. *Peer-to-Peer: Harnessing the benefits of a disruptive technology, O'Reilly*, chapter 8(026115), March 2001.

[82] P. Karbhari, M. Ammar, A. Dhamdhere, H. Raj, G. Riley, and E. Zegura. Bootstrapping in gnutella: A measurement study. In *PAM'04: International Workshop on Passive and Active Network Measurement*, Antibes, France, April 2004.

[83] KaZaA. Kazaa website. http://www.kazaa.com.

[84] J. M. Kleinberg, S. R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. The web as a graph: Measurements, models and methods. In *Proceedings of the International Conference on Combinatorics and Computing*, pages 1–18, 1999.

[85] M. Kleis, E. K. Lua, and X. Zhou. Hierarchical peer-to-peer networks using lightweight superpeer topologies. In *ISCC '05: Proceedings of the 10th IEEE Symposium on Computers and Communications*, pages 143–148, Washington, DC, USA, 27-30 June 2005.

[86] K. Klemm and V. M. Eguíluz. Highly clustered scale-free networks. *Physical Review E*, 65(3):036123, Feb 2002.

[87] P. L. Krapivsky and S. Redner. Organization of growing random networks. *Physical Review E*, 63(6):1–14, May 2001.

[88] P. L. Krapivsky and S. Redner. A statistical physics perspective on web growth. *Computer Networks*, 39:261–276, 2002.

[89] R. Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. S. Tomkins, and E. Upfal. Stochastic models for the web graph. In *Proceedings of the 42nd Annual IEEE Symposium on the Foundations of Computer Science*, pages 57–65, 2000.

[90] K.-W. Kwong and D. H. K. Tsang. Building heterogeneous peer-to-peer networks: Protocol and analysis. *IEEE/ACM Transactions on Networking*, 16(2):281–292, 2008.

[91] D. Leonard, V. Rai, and D. Loguinov. On lifetime-based node failure and stochastic resilience of decentralized peer-to-peer networks. *ACM Sigmetrics Performance Evaluation Review*, 33(1):26–37, 2005.

[92] J. Li, J. Stribling, R. Morris, M. F. Kaashoek, and T. M. Gil. A performance vs. cost framework for evaluating DHT design tradeoffs under churn. In *IEEE INFOCOM*, pages 225–236, Miami, US, 2005.

[93] X. Li and J. Wu. Searching techniques in peer-to-peer networks. In *Handbook of Theoretical and Algorithmic Aspects of Ad Hoc, Sensor, and Peer-to-Peer Networks*, Boca Raton, USA, 2004.

[94] J. Liang, J. Liang, and R. Kumar. Pollution in p2p file sharing systems. In *IEEE INFOCOM*, pages 1174–1185, 2005.

[95] J. Liang, N. Naoumov, and K. W. Ross. The index poisoning attack in p2p file sharing systems. In *IEEE INFOCOM*, Barcelona, Spain, 2006.

[96] J. Lindquista, J. Maa, P. V. D. Driesschea, and F. H. Willeboordsea. Network evolution by different rewiring schemes. *Physica D*, 238(4):370–378, March 2009.

[97] E. López, R. Parshani, R. Cohen, S. Carmi, and S. Havlin. Limited path percolation in complex networks. *Physical Review Letters*, 99(18):188701, Oct 2007.

[98] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim. A survey and comparison of peer-to-peer overlay network schemes. *IEEE Communications Surveys and Tutorials*, 7:72–93, 2005.

[99] E. K. Lua and X. Zhou. Network-aware superpeers-peers geometric overlay network. In *ICCCN 2007: Proceedings of 16th International Conference on Computer Communications and Networks*, pages 141–148, Honolulu, HI, 2007.

[100] Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker. Search and replication in unstructured peer-to-peer networks. In *ICS '02: Proceedings of the 16th international conference on Supercomputing*, pages 84–95, New York, NY, USA, 2002.

[101] C. Magnien, M. Latapy, and J.-L. Guillaume. Impact of random failures and attacks on poisson and power-law random networks. *arXiv:0908.3154v1 [cond-mat.stat-mech]*, 2009.

[102] S. N. Majumdar, M. R. Evans, and R. K. P. Zia. Nature of the condensate in mass transport models. *Physical Review Letters*, 94(180601), 2005.

[103] S. Meng, C. Shi, D. Han, X. Zhu, and Y. Yu. A statistical study of today's gnutella. In *APWeb '06: 8th Asia-Pacific Web Conference, Harbin, China, January 16-18*, pages 189–200, 2006.

[104] P. Merz, M. Priebe, and S. Wolf. Super-peer selection in peer-to-peer networks using network coordinates. In *ICIW '08: Third International Conference on Internet and Web Applications and Services*, June, 2008.

[105] R. Milo, N. Kashtan, S. Itzkovitz, M. E. J. Newman, and U. Alon. On the uniform generation of random graphs with prescribed degree sequences. *eprint arXiv/cond-mat/0312028*, 2003.

[106] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs : Simple building blocks of complex networks. *Science*, 298, 2002.

[107] B. Mitra, N. Ganguly, S. Ghose, and F. Peruani. Generalized theory for node disruption in finite-size complex networks. *Physical Review E*, 78(026115), 2008.

[108] B. Mitra, S. Ghose, and N. Ganguly. Effect of dynamicity on peer to peer networks. In *HiPC'07: Proceedings of the 14th International Conference on High Performance Computing, Goa, India, December 18-21*, pages 452–463, 2007.

[109] B. Mitra, S. Ghose, and N. Ganguly. How stable are large superpeer networks against attack? In *Seventh IEEE International Conference on Peer-to-Peer Computing, September 2-5, Galway, Ireland*, pages 239–242, 2007.

[110] B. Mitra, S. Ghose, N. Ganguly, and F. Peruani. Stability analysis of peer-to-peer networks against churn. *Pramana, Journal of Physics*, 71(02), 2008.

[111] B. Mitra, A. Kumar, S. Ghose, and N. Ganguly. How do superpeer networks emerge? In *IEEE INFOCOM 2010*, San Diego, USA, 2010.

[112] B. Mitra, F. Peruani, S. Ghose, and N. Ganguly. Analyzing the vulnerability of superpeer networks against attack. In *ACM CCS '07: Proceedings of the 14th ACM Conference on Computer and Communications Security.* Alexandria, USA, October 2007.

[113] B. Mitra, F. Peruani, S. Ghose, and N. Ganguly. Analyzing the vulnerability of superpeer networks against attack. In *Proceedings of the 14th ACM Conference on Computer and Communications Security.* Alexandria, USA, October 2007.

[114] B. Mitra, F. Peruani, S. Ghose, and N. Ganguly. Brief announcement: Measuring robustness of superpeer topologies. In *PODC '07: Proceedings of the twenty-sixth annual ACM symposium on Principles of distributed computing*, pages 372–373, Portland, Oregon, USA, 2007.

[115] M. Molloy and B. Reed. A critical point for random graphs with a given degree sequence. *Random Structures and Algorithms*, 6, 1995.

[116] M. Molloy and B. Reed. The size of the giant component of a random graph with a given degree sequence. *Combinatorics, Probability and Computing*, 7, 1998.

[117] A. Montresor. A robust protocol for building superpeer overlay topologies. In *IEEE International Conference on Peer-to-Peer Computing*, pages 202–209, Los Alamitos, CA, USA, 2004.

[118] C. Moore, G. Ghoshal, and M. E. J. Newman. Exact solutions for models of evolving networks with addition and deletion of nodes. *Physical Review E*, 74:3, 2006.

[119] A. E. Motter and Y.-C. Lai. Cascade-based attacks on complex networks. *Physical Review E*, 66:065102, 2002.

[120] A. Nachmias and Y. Peres. Critical percolation on random regular graphs. *Random Structural Algorithms*, 36(2):111–148, 2010.

[121] N. Naoumov and K. Ross. Exploiting p2p systems for DDoS attacks. In *InfoScale '06: Proceedings of the 1st international conference on Scalable information systems*, page 47, New York, NY, USA, 2006.

[122] Netcraft. P2p networks hijacked for ddos attacks. 2007.

[123] M. E. J. Newman. Assortative mixing in networks. *Physical Review Letters*, 20(208701), 2002.

[124] M. E. J. Newman. Spread of epidemic disease on networks. *Physical Review E*, 66(1):016128, Jul 2002.

[125] M. E. J. Newman. Mixing patterns and community structure in networks. *Statistical Mechanics of Complex Networks, R. Pastor-Satorras, J. Rubi and A. Diaz-Guilera (eds.), pp. 66-87, Springer, Berlin*, 2003.

[126] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69(2):026113, Feb 2004.

[127] M. E. J. Newman, S. H. Strogatz, and D. J. Watts. Random graphs with arbitrary degree distributions and their applications. *Physical Review E*, 64(2):026118, Jul 2001.

[128] J. D. Noh. Percolation transition in networks with degree-degree correlation. *Physical Review E*, 76(026116), 2007.

[129] J. Ohkubo, M. Yasuda, and K. Tanaka. Preferential urn model and nongrowing complex networks. *Physical Review E*, 72(6):065104, Dec 2005.

[130] G. Pandurangan, P. Raghavan, and E. Upfal. Building low-diameter p2p networks. In *FOCS '01: Proceedings of the 42nd Annual IEEE Symposium on the Foundations of Computer Science*, pages 492–499, 2001.

[131] K. Park, Y.-C. Lai, and N. Ye. Self-organized scale-free networks. *Physical Review E*, 72(2):026131, Aug 2005.

[132] S.-T. Park, A. Khrabrov, D. M. Pennock, S. Lawrence, C. L. Giles, and L. H. Ungar. Static and dynamic analysis of the internet's susceptibility to faults and attacks. In *IEEE Infocom 2003*, page 31, San-Francisco, CA, USA, 2003.

[133] G. Paul, S. Sreenivasan, and H. E. Stanley. Optimization of network robustness to random breakdowns. *European Physical Journal B*, 38, 2004.

[134] G. Paul, S. Sreenivasan, and H. E. Stanley. Resilience of complex networks to random breakdown. *Physical Review E*, 72(056130), 2005.

[135] G. Paul, T. Tanizawa, S. Havlin, and H. E. Stanley. Optimization of robustness of complex networks. *European Physical Journal B*, 38(2):187–191, March 2004.

[136] V. Paxson. An analysis of using reflectors for distributed denial-of-service attacks. *ACM Computer Communication Review*, 31, 2001.

[137] F. Peruani, A. Deutsch, and M. Baer. Nonequilibrium clustering of self-propelled rods. *Physical Review E*, 74(030904(R)), 2006.

[138] B. Pretre. Attacks on peer-to-peer networks. In *Ph.D thesis*. Swiss Federal Institute of Technology (ETH), Zurich, 2005.

[139] Y. J. Pyun and D. S. Reeves. Constructing a balanced, log(n)-diameter super-peer topology. In *Proceedings of the 4^{th} International Conference on Peer-to-Peer Computing*. Zurich, Switzerland, August 2004.

[140] P. Raftopoulou, E. G. Petrakis, and C. Tryfonopoulos. Rewiring strategies for semantic overlay networks. *Distributed Parallel Databases*, 26(2-3):181–205, 2009.

[141] S. Rhea, D. Geels, T. Roscoe, and J. Kubiatowicz. Handling churn in a DHT. In *ATEC '04: Proceedings of the annual technical conference on USENIX*, pages 10–10, Berkeley, CA, USA, 2004.

[142] M. Ripeanu. Peer-to-peer architecture case study: Gnutella network. In *P2P'01: Proceedings of the First International Conference on Peer-to-Peer Computing*. Sweden, August 27-29 2001.

[143] M. Ripeanu, I. Foster, and A. Iamnitchi. Mapping the gnutella network: Properties of large-scale peer-to-peer systems and implications for system. *IEEE Internet Computing Journal*, 6:2002, 2002.

[144] L. Ronga and I. Burnett. Dynamic resource adaptation in a heterogeneous peer-to-peer environment. In *Second IEEE Consumer Communications and Networking Conference*, pages 416–420, Las Vegas, Nevada, USA, January 2005.

[145] J. Saia. Attack-resistant peer-to-peer networks. In *NIPS 2003 Workshop on Robust Communication Dynamics in Complex Networks*. Whistler, Canada, December 12-13 2003.

[146] S. Saroiu, K. P. Gummadi, and S. D. Gribble. Measuring and analyzing the characteristics of napster and gnutella hosts. *Multimedia Systems*, 9(2):170–184, 2003.

[147] N. Sarshar and V. Roychowdhury. Scale-free and stable structures in complex ad hoc networks. *Physical Review E*, 69:026101, 2004.

[148] H. Schulze and K. Mochalski. Ipoque: Internet study. 2007.

[149] C. Shi, D. Han, Y. Liu, S. Meng, and Y. Yu. A dynamic routing protocol for keyword search in unstructured peer-to-peer networks. *Computer Communication*, 31(2):318–331, 2008.

[150] H. A. Simon. On a class of skew distribution functions. *Biometrika*, 42:425–440, 1955.

[151] A. Singh, M. Castro, P. Druschel, and A. Rowstron. Defending against eclipse attacks on overlay networks. In *EW 11: Proceedings of the 11th workshop on ACM SIGOPS European workshop*, page 21, New York, NY, USA, 2004.

[152] A. Singh, T. W. Ngan, P. Druschel, and D. S. Wallach. Eclipse attacks on overlay networks: Threats and defenses. In *IEEE INFOCOM 2006*, pages 1–12, 2006.

[153] Slashdot. Comcast continues to block peer to peer traffic. 2007.

[154] Slashdot. Ohio university blocks p2p file sharing. 2007.

[155] C. Song, S. Havlin, and H. A. Makse. Origins of fractality in the growth of complex networks. *Nature Physics*, 2(4):275–281, April 2006.

[156] M. Srivatsa, B. Gedik, and L. Liu. Large scaling unstructured peer-to-peer networks with heterogeneity-aware topology and routing. *IEEE Transactions on Parallel and Distributed Systems*, 17(11):1277–1293, 2006.

[157] S. Staniford, V. Paxson, and N. Weaver. How to own the internet in your spare time. In *Proceedings of the 11th USENIX Security Symposium*. Berkeley, CA, August 05-09 2002.

[158] D. Stutzbach and R. Rejaie. Understanding churn in peer-to-peer networks. In *IMC '06: Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, pages 189–202, New York, NY, USA, 2006.

[159] D. Stutzbach, R. Rejaie, and S. Sen. Characterizing unstructured overlay topologies in modern p2p file-sharing systems. In *IMC '05: Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement (USENIX Association)*, pages 5–5, Berkeley, CA, USA, 2005.

[160] D. Stutzbach, R. Rejaie, and S. Sen. Characterizing unstructured overlay topologies in modern p2p file-sharing systems. *IEEE/ACM Transaction on Networking*, 16(2):267–280, 2008.

[161] T. Tanizawa, G. Paul, R. Cohen, S. Havlin, and H. E. Stanley. Optimization of network robustness to waves of targeted and random attacks. *Physical Review E*, 71(056130), 2005.

[162] T. Tanizawa, G. Paul, S. Havlin, and H. E. Stanley. Optimization of the robustness of multimodal networks. *Physical Review E*, 74(1):016125, July 2006.

[163] K. Truelove. To the bandwidth barrier and beyond. http://web.archive.org/web/20010107185800/dss.clip2.com/gnutella.html. 2000.

[164] A. Valente, A. Sarkar, and H. Stone. 2-peak and 3-peak optimal complex networks. *Physical Review Letters*, 92(118702), 2004.

[165] A. Vazquez and Y. Moreno. Resilience to damage of graphs with degree correlations. *Physical Review E*, 67(015101 R), 2003.

[166] B. Wanga, H. Tanga, C. Guoa, Z. Xiub, and T. Zhouc. Optimization of network structure to random failures. *Physica A: Statistical Mechanics and its Applications*, 368(2):607–614, 2006.

[167] B. Wilcox-O'Hearn. Experiences deploying a large-scale emergent network. In *IPTPS '01: Revised Papers from the First International Workshop on Peer-to-Peer Systems*, pages 104–110, London, UK, 2002.

[168] R. Wouhaybi and A. Campbell. Building resilient low-diameter peer-to-peer topologies. *Computer Networks*, 52(5):1019–1039, December 2007.

[169] R. H. Wouhaybi and A. T. Campbell. Phenix: supporting resilient low-diameter peer-to-peer topologies. In *INFOCOM 2004, March*, page 119, Hong Kong, 2004.

[170] B. Yang and H. Garcua-Molina. Designing a super-peer network. In *ICDE '03: Proceedings of the International Conference on Data Engineering*. Los Alamitos, CA, March 2003.

[171] Z. Yao, D. Leonard, X. Wang, and D. Loguinov. Modeling heterogeneous user churn and local resilience of unstructured p2p networks. In *ICNP '06: Proceedings of the IEEE International Conference on Network Protocols*, pages 32–41, Washington, DC, USA, 2006.

[172] L. Zhao, K. Park, and Y.-C. Lai. Attack vulnerability of scale-free networks due to cascading breakdown. *Physical Review E*, 70(3):035101, Sep 2004.