Probabilistic models and efficient algorithms for certain large-scale distributed networks

Zeng, Jianyang

2007

Zeng, J. Y. (2007). Probabilistic models and efficient algorithms for certain large-scale distributed networks. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/35731>

<https://doi.org/10.32657/10356/35731>

Nanyang Technological University

**Probabilistic Models and Efficient Algorithms
for Certain Large-Scale Distributed Networks**

by

Jianyang ZENG

School of Computer Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirements of the degree of
Doctor of Philosophy

2007

Abstract

Two common topological features have been empirically reported in real-world large-scale distributed networks, such as the Internet, Web and peer-to-peer networks. They are, namely, (a) the small-world phenomenon where any pair of nodes in the network can be connected by a short chain of intermediates, and (b) the scale-free connectivity pattern where the node degrees follow a power law distribution. Modelling these distributed networks can help understand their growth mechanisms and exploit other useful characteristics for applications. However, most existing models are far from being satisfactory in that they all depend on unrealistic assumptions, e.g., each node is supposed to know about the entire network. My research aims to develop effective models that are more suitable for constructing real-world distributed systems.

In this thesis, I propose a new model for the constructions of scale-free networks. Each new joining node would use only a small sample set of existing nodes to establish its connections. I demonstrate that the power law of degree distribution still holds true even if each new joining node knows only a logarithmic number of existing nodes. Compared with the existing models which require each node to process global information, the construction based on partial information better approximates the evolution of many scale-free networks arising in the real world. The result may also be applied in the constructions of peer-to-peer networks, where global information is generally unavailable for any individual node.

In order to address an outstanding routing issue in the small-world networks, I extend Kleinberg's original model by augmenting the local links for each node. Based on the extended model, I propose both near-optimal and optimal decentralized algorithms for message routing. The near-optimal algorithm is able to route messages between any pair of nodes in $O(\lg n \lg \lg n)$ expected number of hops, and requires only $O(\lg^2 n)$ bits of information stored on each node, where n denotes the network size. The optimal decentralized routing algorithm performs in $O(\lg n)$ expected number of hops and requires no

more than poly-logarithmic number of bits stored on each node. Given the same amount of information stored on each node, the near-optimal algorithm outperforms the existing schemes in Kleinberg's small-world networks. The optimal algorithm, to the best of my knowledge, is the first to achieve the optimal routing complexity while still keeping a poly-logarithmic number of bits of information on each node. My investigation has shown that the augmented local connections indeed make small-world networks more navigable. The results may be applied to improve the performance of current peer-to-peer systems.

The third contribution of my thesis lies in the design and analysis of ReCord, a recursive distributed hash table for peer-to-peer networks. It is a generalized version of Randomized-Chord which offers improved tradeoffs in performance and topology maintenance over existing peer-to-peer systems. An n -node ReCord with $O(\lg n)$ node degree can achieve an expected latency of $O(\lg n)$ hops. Alternatively, it can also offer $O(\frac{\lg n}{\lg \lg n})$ hops latency at a higher cost of $O(\frac{\lg^2 n}{\lg \lg n})$ node degree. ReCord provides an adaptable infrastructure which adjusts the real-time network connections by monitoring the peer-to-peer system dynamically.

In summary, the results have provided novel perspectives for the constructions of large-scale distributed networks. I am confident that the new insights will help better understand the evolutions of real-world distributed networks and contribute to their further applications.

Acknowledgments

First and foremost, I thank my advisor Dr. Hsu Wen-Jing for his guidance, encouragement and help during the past three years. It was he who brought me to the wonderful field of algorithms. It was his patient supervision and thoughtful advice that made it possible for me to develop whatever research capabilities I have exhibited. The research in this thesis was all produced as a joint effort with him. Without his research input, constructive comments and careful revisions on my draft papers or thesis, the contributions in this thesis would have been much less substantial. I would like to express my deepest appreciation to him.

I would like to thank all team members of HCTS project (High Capacity Terminal Simulation System for Handling Mega Container Vessels), especially Stuti and Vee Voon Yee (Dr) for their friendship and useful advice on improving my oral and written English.

I warmly thank my colleagues in our parallel and distributed research group: Hu Yahong (Dr), He Yuxiong, Chung Shin Yee, Fang Hui, Seet Boon Chong. Discussing the research problems with them has always been enjoyable.

I would like to thank other members in Center for Advanced Information Systems (CAIS), Nanyang Technological University, especially Ye Rong, Chen Chuanyu, Qiu Ling (Dr), Zhao Qiankun, Li Zhuowei, Weng Jianshu, Gong Ruibin, Liu Zehua, Wang Yida, Sun Aixin (Dr), Li Dongtao, Sun Zhen, etc., for their friendship and all the good times.

I devote my special appreciation to my girl friend, Wang Jiangdian, for her love, care, support, and encouragement. She has been an important person in my life.

I would also like to thank Lay Choo and Chee Keong, technicians of Center for Advanced Information Systems (CAIS), for maintaining a comfortable computing environment.

Last but certainly not the least, I would like to thank my family for their constant emotional support during the past years.

Contents

Abstract	i
Acknowledgments	iii
List of Figures	viii
List of Tables	ix
1 Introduction	1
1.1 Background and Motivations	1
1.2 Objectives	3
1.3 Approaches and Results	3
1.3.1 Evolution of Scale-Free Networks with Partial Information	4
1.3.2 Near-optimal Routing for Small-World Networks with Augmented Local Awareness	4
1.3.3 Optimal Routing for Small-World Networks with Augmented Local Awareness	5
1.3.4 Experimental Study of Small-World Networks and Applications to Peer-to-Peer Systems	5
1.3.5 ReCord: A Distributed Hash Table with Recursive Structure	5
1.4 Synopsis of Thesis	6
2 Models of Large-Scale Networks	7
2.1 Introduction	7
2.2 Small-World Models	8
2.2.1 Current Popular Small-World Models	8
2.2.2 Kleinberg's Small-World Model	11
2.3 Scale-Free Networks	13

CONTENTS

2.4	Peer-to-Peer Overlay Networks	16
2.4.1	Characteristics of Peer-to-Peer Systems	16
2.4.2	Design Issues in Peer-to-Peer Systems	18
2.4.3	Network Structures	19
2.4.4	Comparisons of Different Structures	24
2.5	Relations between Social and Peer-to-Peer Networks	24
2.5.1	Observations of Social Behaviors from Real-World Peer-to-Peer Networks	25
2.5.2	Applications of Social Studies to the Peer-to-Peer Network Design	26
2.5.3	Incentive Mechanisms for Peer-to-Peer Networks	27
2.6	Prospective Research	28
3	Evolution of Scale-Free Networks with Partial Information	30
3.1	Introduction	30
3.2	The Model and Notations	32
3.3	Our Main Results	34
3.4	Mathematical Analysis	35
3.4.1	The Degree Distribution	35
3.4.2	Maximum Degree	44
3.4.3	Diameter	46
3.5	Experimental Evaluation	47
3.6	Related Work	48
3.7	Concluding Remarks	48
4	Near-Optimal Routing for Small-World Networks with Augmented Local Awareness	52
4.1	Introduction	53
4.1.1	Our Contributions	54
4.2	Definitions of Small-World Models	55
4.3	Decentralized Routing Algorithms	57
4.4	near-optimal Routing with $O(\lg n)$ Awareness	58
4.4.1	Primary Augmented Local Awareness of $O(\lg n)$	58

CONTENTS

4.4.2	Non-Oblivious Decentralized Routing	60
4.4.3	Oblivious Decentralized Routing	63
4.5	Concluding Remarks	67
5	Optimal Routing in a Small-World Network	68
5.1	The Small-World Model and Decentralized Routing	69
5.2	Our Contributions	70
5.3	Analysis of Decentralized Routing	72
5.4	Concluding Remarks	80
6	Experimental Study of Small-World Networks and Applications to Peer-to-Peer Systems	82
6.1	Experimental Study of Routing in Kleinberg's and Extended Small-World Networks	83
6.1.1	Experimental Setup	83
6.1.2	Experimental Results	84
6.2	Improved Symphony with Augmented Local Awareness	85
6.2.1	Construction of Symphony*	85
6.2.2	Evaluation of Routing Performance	87
6.2.3	Evaluation of Resilience to Link Failure	87
6.3	Routing in Watts-Strogatz's Small-World Networks	89
6.3.1	Watts-Strogatz's Small-World Network	89
6.3.2	Experimental Setup	90
6.3.3	Performance Metrics	90
6.3.4	Experimental Results	91
6.4	Network Size Estimation for Non-Uniform Node Distribution	92
6.4.1	Our Sampling Algorithms	93
6.4.2	Data Generation for Testing	94
6.4.3	Experimental Setup	95
6.4.4	Experimental Results	96
6.5	Concluding Remarks	97

CONTENTS

7	ReCord: A Distributed Hash Table with Recursive Structure	104
7.1	Introduction	104
7.2	Construction of ReCord	106
7.3	Analysis	107
7.4	Implementation of ReCord	110
7.4.1	Estimation of network size	110
7.4.2	Join and Leave maintenance	111
7.4.3	Re-linking operation	112
7.4.4	Fault tolerance	113
7.5	Experiments	114
7.5.1	Estimation of network size	114
7.5.2	Degree and Latency	116
7.5.3	Fault tolerance	116
7.6	Concluding Remarks	117
8	Discussions and Conclusion	119
8.1	Summary of Contributions	119
8.2	Future Work	120
8.3	Concluding Remarks	122
	Appendix A Pseudocodes for ReCord Implementation	124
	References	126

List of Figures

3.1	The distribution of node degree with different sizes of sample set.	50
3.2	Maximum node degree in networks with different numbers of nodes.	51
4.1	Diagram for oblivious decentralized routing.	66
6.1	Average length of routing paths for different routing schemes.	85
6.2	Storage requirement on each node for different routing schemes.	86
6.3	Average length of routing path in the original Symphony and Symphony* .	87
6.4	Cost of node joining/leaving in the original Symphony and Symphony* . .	88
6.5	Comparison of fault tolerance between Symphony and Symphony*	88
6.6	Average $D(p)/D(0)$, $C(p)/C(0)$ and $R(p)/R(0)$ for different n	98
6.7	Average $D(p)/D(0)$, $C(p)/C(0)$ and $R(p)/R(0)$ for different k	99
6.8	Examples of data generated for testing (total number of nodes=1000). . . .	100
6.9	Network size estimation with different choices of M_2	101
6.10	Network size estimation for different choices of N	102
6.11	Network size estimation for different choices of w	103
7.1	An example of interval division ($k = 4$).	107
7.2	Impact of variable k on degree and latency.	110
7.3	Estimation of network size for stable P2P systems.	115
7.4	Estimation of network size for expanding, shrinking, and stable network respectively.	115
7.5	Average degree and latency on expanding network.	116
7.6	Tradeoff between degree and latency over different choices of k in a stable network (the number of active nodes is between [2070,2121]).	117
7.7	Impact of link failures on the average latency.	118

List of Tables

2.1	Comparisons of current small-world models.	11
2.2	Performance comparisons of existing decentralized routing algorithms in Kleinberg’s small-world model.	12
2.3	Comparisons of different peer-to-peer structures.	24
4.1	Comparisons of our decentralized routing algorithms with other existing schemes.	55
5.1	Comparisons of our decentralized routing algorithms with the other existing schemes.	72
6.1	Experimental parameters for estimating the network size.	95

Chapter 1

Introduction

1.1 Background and Motivations

Peer-to-peer (P2P) networks have become popular in resource-sharing applications recently. There have been millions of users in certain successful systems, such as Napster [79], Gnutella [42], and Kazaa [59]. P2P systems are distributed systems without any central authority (or hierarchical organization), and each node in the system performs similar functions.

Although the peer-to-peer network topology largely depends on the artificial protocols designed by computer engineers, it also exhibits certain ad hoc behaviors of distributed users, and thus displays interesting social characteristics, as similarly observed in the social networks [56, 53, 66]. In this sense, the peer-to-peer network may be viewed as a “natural” network to some degree, like other networks arising from the Internet, Web, society, biology, etc. This intuition suggests that we may adapt the research approaches from other disciplines, such as physics and sociology, to peer-to-peer systems. A well-known example of such multi-disciplinary research is the finding by researchers from social studies that the popularity of web pages follows a Zipf distribution, which is then exploited to improve caching schemes by the computer scientists [19].

In [27, 10], it has been empirically observed that many real-world complex networks, such as Internet, Web, social networks, etc., display certain common characteristics, namely,

CHAPTER 1. INTRODUCTION

Small-World phenomenon [27] and Power Law distribution of node degrees [10]. Informally, the small-world phenomenon is the hypothesis that everyone can reach any other in the world by a short chain of acquaintances. In a network with power-law degree distribution, most nodes are with small node degrees, and only a small number of nodes have relatively large degrees. These two phenomena are also found to exist in some current peer-to-peer systems, such as Gnutella [42], Kazaa [59], etc. The universality of these two phenomenon induces the research topics on how they are formed, and how the study can be applied to computer science. The first step is to theoretically model the formation or evolution of the real-world networks displaying the small-world or/and power law effect(s). A mathematical model is useful for studying the properties of the real-world complex networks such as the node degree distribution [18], connectivity [77], maximum degree [25], diameter [17], etc. Knowledge concerning the structures of the networks and the evolution mechanisms of these structures may find applications in many contexts, such as overlay network constructions [88], virus attacks [16, 54], etc.

Unfortunately, although the existing models are consistent with the observation of real-world properties, most of them are based on very idealized assumptions. For example, the well known Barabási-Albert model for complex networks with power law distribution of node degree assume that a newly added node knows all existing nodes, such as their node degrees [10]. This is an unrealistic assumption with real-world large-scale networks such as the Internet or peer-to-peer networks, as it requires a node to access and process an extremely large amount of information. Thus, it would be very useful to model the real-world networks with desired properties while best fitting the evolution track of the practical networks.

Another challenge is to apply the theoretical models to the design of efficient peer-to-peer networks. One successful example has been demonstrated in [106], where the principle of small-world model is used to improve Freenet's performance; Symphony [70] is another

example that uses Kleinberg's small-world model to design a Dynamic Hash Table (DHT) for peer-to-peer systems.

1.2 Objectives

The objectives and goals of our research are as follows.

- *To propose effective models for the real-world networks with small-world and power law effects.* The effective models should closely approximate the evolution of real-world networks, and help us better understand the emergence of their characteristics and hence are useful for the system development.
- *To design efficient peer-to-peer systems by applying the principles of the models of real-world networks.* We aim to improve Kleinberg's small-world and apply it to the design of peer-to-peer networks. Our objective is to develop efficient routing algorithms with short latency and low space requirement for the small-world model. The improvement of Kleinberg's small-world models will help us design more efficient distributed routing algorithms.
- *To design efficient Dynamic Hash Tables (DHTs) for peer-to-peer overlay networks supporting fast content searching and with low maintenance overhead.* We aim to propose a generalized version of Randomized-Chord [70], offering improved tradeoffs in performance and topology maintenance over existing P2P systems. The generalized version of Randomized-Chord will give us some hints to design more flexible and efficient peer-to-peer systems.

1.3 Approaches and Results

The main contributions of this thesis are described in the following subsections.

1.3.1 Evolution of Scale-Free Networks with Partial Information

We will propose a new model, in which each new joining node uniformly and randomly chooses a sample set of existing nodes, and then connects to some nodes in the sample set according to certain rules, for which the power law of degree distribution holds true even if each new joining node knows only a logarithmic number of existing nodes. Compared with the existing models which require each node to know about the entire network, our construction based on partial information will better approximate the evolution of many known networks arising in the real world. For instance, our models and results may be applied to the constructions of large-scale distributed systems such as peer-to-peer networks, where the global information is generally unavailable for any individual node in the network.

1.3.2 Near-optimal Routing for Small-World Networks with Augmented Local Awareness

We will extend Kleinberg's small-world model and add three extra local links for each node x . Our investigation aims to learn how these augmented local connections can make small-world networks more navigable.

There are normally two approaches for decentralized routing: oblivious and non-oblivious schemes [39]. A routing algorithm is *oblivious* if the message holder makes routing decisions only based on its own routing table and the target node. On the other hand, a routing algorithm is said to be *non-oblivious* if the routing decisions of the message holder depend on the previous routing history stored in the message header, the target node and its routing table. We will show that if each node is aware of $O(\lg n)$ number of neighbors via the augmented local links, there exist both non-oblivious and oblivious algorithms that can route messages between any pair of nodes in $O(\lg n \lg \lg n)$ expected number of hops, which outperforms the existing results for Kleinberg's small-world networks. Our schemes

keep only $O(\lg^2 n)$ bits of routing information on each node. Thus our schemes are scalable with respect to the network size.

1.3.3 Optimal Routing for Small-World Networks with Augmented Local Awareness

We will extend Kleinberg’s small-world model in that each node x has two more random links. We are the first to achieve the optimal routing complexity while still keeping a poly-logarithmic number of bits of information stored on each node in the small-world networks. Our results may be applied to the design of the logical overlay structure of large-scale distributed systems, such as peer-to-peer networks, as with Symphony [70]. Since the links are randomly constructed according to the probabilistic distribution, our extended small-world network is less vulnerable to adversarial attacks, and thus provides good fault tolerance.

1.3.4 Experimental Study of Small-World Networks and Applications to Peer-to-Peer Systems

We will experimentally study the routing issues in some small-world models, including existing routing schemes in Kleinberg’s models, our two versions of routing schemes in the extensions of Kleinberg’s models, and the greedy routing in Watts-Strogatz’s models. Our simulations will verify our theoretical results in the extended versions of Kleinberg’s small-world model. We will also investigate certain issues about deploying small-world overlay structures for peer-to-peer systems, such as the improvement of routing in Symphony [70], a well-known peer-to-peer overlay network with small-world-like connections.

1.3.5 ReCord: A Distributed Hash Table with Recursive Structure

We will investigate a generalized version of Randomized-Chord and offers improved trade-offs in performance and topology maintenance over existing P2P systems. As a result, a

CHAPTER 1. INTRODUCTION

simple network called ReCord is proposed. It is scalable and can be easily implemented as an overlay network, and offers a good tradeoff between the node degree and query latency. For instance, an n -node ReCord with $O(\lg n)$ node degree has an expected latency of $\Theta(\lg n)$ hops. Alternatively, it can also offer $\Theta(\frac{\lg n}{\lg \lg n})$ hops latency at a higher cost of $O(\frac{\lg^2 n}{\lg \lg n})$ node degree. Simulations of the dynamic behaviors of ReCord will also be carried out.

1.4 Synopsis of Thesis

The remainder of the thesis is organized as follows. In Chapter 2, we present a comprehensive survey of related work. In Chapter 3, we propose a evolution model for scale-free networks with partial information. In Chapter 4, we present an extended version of Kleinberg's small-world model and present near-optimal algorithms for routing messages in this model. In Chapter 5, we describe an optimal routing algorithm in an extended small-world model with augmented local awareness. In Chapter 6, we experimentally study the routing issues in the current small-world models. In Chapter 7, we present a distributed hash table with recursive structure. In Chapter 8, we briefly summarize the research and discusses possible future work.

Chapter 2

Models of Large-Scale Networks

In this chapter, we will review related work on models for large-scale networks.

Section 2.1 gives a brief introduction. Section 2.2 reviews a few popular small-world models. Section 2.3 reviews models of scale-free networks. Section 2.4 gives a brief survey of current peer-to-peer networks, including both structured and unstructured networks. In Section 2.5, we examine the relations between the social networks and peer-to-peer systems. Section 2.6 describes prospective research directions.

2.1 Introduction

Towards the end of the last century, a bulk of research has been carried out for the modelling of two interesting phenomena in the Internet or Web graphs: small-world and power law degree distribution [74, 3, 98, 29, 80]. Besides helping us understand how real-world networks evolve, the models of these complex networks can also be applied to the construction of large-scale distributed systems. One example is the peer-to-peer networks, which are logical virtual overlays built over the Internet topology, and have found many applications in resource sharing [7].

In the following sections, we will give a brief review on current small-world and scale-free models, and also survey some popular peer-to-peer networks, including both structured and unstructured ones. We then review some relations between the social networks and

peer-to-peer systems, including the observation of social networks in the existing peer-to-peer systems, applications of social principles to the peer-to-peer network design, and the incentive mechanisms for peer-to-peer systems. We believe that exploring these aspects would help give useful insight for the development of more efficient peer-to-peer networks.

2.2 Small-World Models

2.2.1 Current Popular Small-World Models

Most early work was dedicated to explaining why random networks have small diameters [32]. A well-known study by Milgram in 1967 [93] reveals the *small-world phenomenon* [32], also called “six degree of separation”, that any two people in the world can be connected by a chain of six (on the average) acquaintances, and people can deliver messages efficiently to an unknown target via this social network of acquaintances. This study is repeated by Dodds, Muhamad, and Watts [28] recently, and the results show that it is still true for today’s social network. The small-world phenomenon has also been shown to permeate various types of networks arising from nature and engineering systems; examples include the World Wide Web [27, 4], peer-to-peer systems [8, 71, 70, 106], etc.

Recently, a number of network models have been proposed to study the small-world properties [74, 27, 61]. Watts and Strogatz [27] first introduce the concept of “small-world networks”, which are loosely defined as networks with low diameters (e.g. having a logarithmic number of hops between any two nodes) and high clustering coefficient. Compared with conventional random graphs [83] that arise more from theoretical interests, the small-world networks are motivated from an observation that, instead of being completely regular or completely random, the topologies of real-world networks are somewhere in between these two extremes. Watts and Strogatz present a “rewiring procedure” to interpolate between regular and random connections. Their model starts with an n -node ring enhanced with k links on each node to the node’s closest neighbors along the ring. Then with probability

CHAPTER 2. MODELS OF LARGE-SCALE NETWORKS

p , each link is reconnected to another node chosen randomly and uniformly over the entire ring, whereas with probability $1 - p$, each link remains unchanged. Watts and Strogatz show that there is a range of p such that the network displays the small-world effect, that is, the network has small average diameter and large clustering coefficient, which is much larger than that of the conventional random graphs. Most recent work on the small-world is based on Watts-Strogatz's model [74]. Additional properties of small-world networks, such as the scaling function of vertex-vertex distance, have been explored in [75, 76, 92]; the eigenspectrum of the Laplacian operator on small-world graph is investigated in [86], etc.

An alternative view to explain the small-world phenomenon is that they all have relatively few nodes in the network which are connected to a widely distributed set of the other nodes [58]. Another small-world model is proposed by Albert et al. [4] based on an empirical study of the World Wide Web. They claim that the Web is dominated by a small set of nodes with high connections, and the distribution of node degree follows a power law. In both models mentioned above, most nodes are connected via a small number of "important" nodes so that there exists a short path between any two nodes. We refer to these models as *Elitist small-world model*. Although the models contain efficient routing paths, the elites also incur load imbalance. For example, any two nodes in a star graph can be connected by a routing path with just two hops, but the node at the center can be highly loaded.

All the preceding small-world models have taken a small diameter to the equivalent to efficient routing. This equivalence is true only if each node is aware of all the nodes in the network and can compute the shortest path to any destination. However, in many large scale networks, each node can only gather limited, localized information. A routing path discovered by using only local information may not result in the shortest path. As noted by Kleinberg [61], the small (poly-logarithmic) diameter of some graphs does not imply

CHAPTER 2. MODELS OF LARGE-SCALE NETWORKS

the existence of efficient distributed routing algorithms. For instance, the random graph in [9] has a logarithmic diameter, yet any routing using only local information requires *at least* $\sqrt{n}$ expected number of hops (where n is the size of the network) [61]. Therefore, it is of significance to propose a small-world model in which an efficient routing path can be found by using only local information. Kleinberg [61, 62] also argues that Watts-Strogatz's model, in which shortcuts connect nodes at any distance with uniform probability, may not reflect the real world.

In order to address the routing aspects, Kleinberg develops a new model based on a d -dimensional torus with long-range links chosen randomly from the d -harmonic distribution. Based on this model, Kleinberg then shows that routing messages between any two nodes can be achieved in $O(\lg^2 n)$ expected number of hops by applying a simple greedy routing algorithm using only local information. Kleinberg also examines other models in which the long-range links are established according to the r -harmonic distribution, where $r \neq d$. Kleinberg shows that efficient routing paths generally do not exist in these variant models. This finding implies that the long-range contacts chosen by the d -harmonic distribution provide the best clues for finding the shortest path.

In Kleinberg's model, the probability for a long-range link to exist between nodes u and v is proportional to $\frac{1}{\text{Dist}(u,v)^d \cdot \lg n}$, where $\text{Dist}(u,v)$ denotes the Manhattan distance between nodes u and v . Thus, the (global) information of the network size is required before establishing the long-range link. Another shortcoming of Kleinberg's model is that it requires each node to know the underlying structure that is d -dimensional torus, which may not be practical at all.

A comparison of the existing small-world models is shown in Table 2.1.

Applications of the small-world phenomenon in computer networks include efficient lookups in peer-to-peer systems [71, 8, 70, 106], gossip protocols in a communication network [60], flooding routing in ad-hoc networks [48], and the study of diameter of World Wide Web [4], etc.

CHAPTER 2. MODELS OF LARGE-SCALE NETWORKS

Models	Description	High clustering coefficient?	Efficient distributed routing?	Balanced load?	Need global information for the nodes to form the network?
Watts-Strogatz's model [27]	Start with a n -node ring with each node connected to k closest neighbors. Each link is rewired to a random node with probability p .	Yes	Unknown	Yes	No
Elitist model [58] [4]	Most nodes are connected via a small number of highly connected nodes.	Unknown	Yes	No	Yes
Kleinberg's model [61]	Based on a d -dimensional torus with long-range links chosen randomly from the d -harmonic distribution.	Unknown	Yes	Yes	Yes

Table 2.1: Comparisons of current small-world models.

2.2.2 Kleinberg's Small-World Model

In order to incorporate routing or navigating properties into probabilistic graph models, Kleinberg [61, 62] investigates a family of network models which also naturally generalize Watts and Strogatz's model, and reports that one of the models allows messages to be routed efficiently by using only local information. This model is based on a d -dimensional grid strengthened by adding random long-range links to each node according to the d -harmonic distribution, as mentioned in the preceding subsection. Kleinberg shows that the route between any two nodes can be found by using a simple decentralized routing algorithm in $O(\lg^2 n)$ expected number of hops. This bound is tightened to $\Theta(\lg^2 n)$ later by Barrière et al. [11] and Martel et al. [72]. Further research [71, 65, 72, 39] shows that in fact the $O(\lg^2 n)$ bound of the original greedy routing algorithm can be improved by embedding some extra information in each message holder. Manku, Naor, and Wieder [71] show that if each message holder at a routing step takes its own neighbors' neighbors into account for making routing decisions, the bound of routing complexity can be improved to $O(\frac{\lg^2 n}{q \lg q})$, where q denotes the number of long-range contacts. Lebhar and Schabanel [65] also propose a routing algorithm for Kleinberg's 1-dimensional model, which visits $O(\frac{\lg^2 n}{\lg^2(1+q)})$ nodes on expectation before routing the message, and they show that a routing path with expected length of $O(\frac{\lg n (\lg \lg n)^2}{\lg^2(1+q)})$ can be found. Two research groups, Fraigniaud et al. [39],

CHAPTER 2. MODELS OF LARGE-SCALE NETWORKS

Scheme	#bits of awareness	#steps expected	Oblivious or Non-oblivious?
Kleinberg's greedy [61, 8, 11]	$O(q \lg n)$	$O(\lg^2 n / q)$	Oblivious
NoN-greedy [71]	$O(q^2 \lg n)$	$O(\lg^2 n / (q \lg q))$	Non-oblivious
Decentralized algorithm in [65]	$O(\lg^2 n / \lg(1 + q))$	$O((\lg n)^2 / \lg^2(1 + q))$	Non-oblivious
Decentralized algorithm [72]	$O(\lg^2 n)$	$O((\lg n)^{1+1/d})$	Non-oblivious
Indirect-greedy algorithm [39]	$O(\lg^2 n)$	$O((\lg n)^{1+1/d})$	Oblivious

Table 2.2: Performance comparisons of existing decentralized routing algorithms in Kleinberg's small-world model.

and Martel and Nguyen [72], independently report that if each node is aware of $O(\lg n)$ of its closest local neighbors, the routing complexity in Kleinberg's d -dimensional small-world networks can be improved to $O(\lg n \lg^{1+1/d} n)$ expected number of hops. The difference is that [72] requires keeping additional state information, while [39] uses an oblivious greedy routing algorithm. Similarly, Fraigniaud et al. [39] show that $\Theta(\lg^2 n)$ bits of topological awareness per node is optimal for their oblivious routing scheme. In [72], Martel and Nguyen also show that the expected diameter of a d -dimensional Kleinberg network is $\Theta(\lg n)$. In other words, there is still some room for reducing the routing complexity, which motivates our work. Table 2.2 shows a comparison of the existing decentralized routing algorithms for Kleinberg's small-world model.

Other small-world models have also been studied. In their recent paper [82], Nguyen and Martel study the diameters of several variants of Kleinberg's small-world models, and they provide general guidelines for constructing classes of small-world networks with $\Theta(\lg n)$ expected diameter. Aspnes, Diamadi, and Shah [8] find that the greedy routing algorithms in directed rings with a constant number of random extra links given in *any* distribution requires at least $\Omega(\lg^2 n / \lg \lg n)$ expected number of hops. Another related models are the small-world percolation models [71, 51, 26, 68]. The diameters of these models are studied respectively by Benjamin et al. [51], Coppersmith et al. [26] and Biskup [68]. The routing aspects of the percolation models, such as the lower bound and upper bound of greedy 1-lookahead routing algorithms, are studied in [71]. All these models provide alternative methods to construct small-world networks with short diameters.

2.3 Scale-Free Networks

Recently, a great amount of attention has been focused on the networks that exhibit power law degree distribution [3, 98, 29, 80], that is, the proportion P_k of nodes with degree k follows a power law distribution: $P_k \sim k^{-r}$. Because the exponent r in the power law is a constant independent of the network size, such type of network is referred to as a *scale-free* network.

It has been reported empirically by several researchers, e.g. in [4, 20, 35], that both the Internet and the Web have the *scale-free* property: The proportion P_k of nodes with degree k follows a power law distribution: $P_k \sim k^{-r}$, where $r \leq 3$. Many other real-world networks, such as citation networks [85], metabolic networks [46], movie actors [80], etc., also exhibit similar properties.

The conventional random graphs models introduced by Erdős and Rényi [83] do not yield the power law distribution, and they require modifications for modelling and analyzing these large-scale networks. A survey on the mathematical models for the scale-free networks can be found in [14]. Two comprehensive surveys (by physicists) can be found in [3, 29].

A popular model for producing the power law or scale-free phenomenon is the so-called ‘BA’ model named after its inventors Barabási and Albert [10]. The BA model is based on a preferential attachment (PA) rule, i.e., a new node is added into the network at each time step, and the probability that an existing node x gets a link from the new node is proportional to x ’s degree. Here we quote the construction of the BA model from [10]:

...Starting with a small number (m_0) of vertices, at every time step we add a new vertex with m ($\leq m_0$) edges that link the new vertex to m different vertices already present in the system. To incorporate preferential attachment, we assume that the probability Π that a new vertex will be connected to a vertex i depends on the connectivity k_i of that vertex, so that $\pi(k_i) = k_i / \sum_j k_j$. After t steps the model leads to a random network with $t + m_0$ vertices and mt edges.

CHAPTER 2. MODELS OF LARGE-SCALE NETWORKS

However, the BA model is not mathematically sound [18]. Bollobás and Riordan refine the BA model and make it more precise and theoretically analyzable [17, 18]. They find that although the BA model is a dynamic stochastic process, it can be regarded as a “LCD (Linearized Chord Diagrams)” [17], which is a static problem and is easier to solve. The result in the LCD model gives a rigorous justification of the power law distribution of the node degrees in the BA model.

Many variations of the preferential attachment rule originated from the BA model have been developed [17, 18, 30, 31, 21, 25]. A variation on the BA model, in which each node has an “initial attractiveness” has been introduced independently by two different groups, Dorogovtsev et al. [30] and Drinea et al. [31]. A precise and rigorous analysis of this model is given by Buckley and Osthus [21]. Cooper and Frieze [25] present a general model which considers more parameters, such as a variable number of edges or nodes generated at each time step, a mixture of uniform and PA selection, etc.

Aiello, Chung, and Lu [2] introduce a scale-free model with a prescribed degree sequence, where the probability of two nodes being connected is proportional to the product of their expected degrees, which are given in advance. Although their model is consistent with the power law distribution, it can not be used to explain how power law arises.

Kumar et al. [64] independently presented a so-called *copying model*, motivated by the fact that a new web node is often generated by copying an old one and changing some of its old links. Their copying model is based on the following observation. When an author is going to create a new web page, he usually has some topic in his mind. A new viewpoint about the topic is likely to link to some existing pages about the topic, but also probably connect to some new pages unrecognized before. The degree distribution of the copying model follows a power law. Another property is that there exist many bipartite subgraphs in the resulted network. Indeed, the original motivation of the copying model is to explain the appearance of bipartite subgraph phenomenon in the web graph.

CHAPTER 2. MODELS OF LARGE-SCALE NETWORKS

Fabrikant, Koutsoupias, and Papadimitriou (FKP) [34] propose a “Highly Optimized Tolerance” (HOT) framework to model the Internet network, i.e., the connection of the new node is constructed according to an adjustable neighborhood. In FKP model, each node is mapped into a random point in a unit square. The model starts with a point x_0 uniformly and randomly chosen in the unit square, and set $W(x_0) = 0$. Next, for the sequence of points $1, 2, \dots, n$, a new node x_i is chosen uniformly and randomly from the unit square, and is connected to a previous point x_j to minimize the

$$W(x_j) + \alpha D(x_i, x_j),$$

where α is the model parameter, and $D(x_i, x_j)$ denotes the Eulidean distance between points x_i and x_j . If there is a link between x_i and x_j , then $W(x_i) = W(x_j) + 1$. The authors in [34] have proved a power law lower bound for part of the degree sequence given a range of parameter α . However, as stated by Berger et al. [12], the power law in FKP model is very different from that in the previous models. Indeed, the overall degree distribution is far from a power law: the power law phenomenon exists only for a small-fraction of the nodes. Berger et al. [13] propose an extended version of FKP model, called competition-induced preferential attachment model.

Because of its significance, the construction of their model is quoted as follows [13].

Let A_1, A_2 be two positive integer-valued parameters. Let $G(t)$ be the following Markov process, whose states are finite rooted trees in which each node is labelled either fertile or infertile. At time $t = 0$, $G(t)$ consists of a single fertile vertex. Given the graph at time t , the new graph is formed in two steps: first, a new vertex, labelled $t + 1$ and initialized as infertile, connects to an old vertex j with probability zero if j is infertile, and with probability

$$\Pr[t + 1 \rightarrow j] = \frac{\min\{d_j(t), A_2\}}{W(t)},$$

if j is fertile. Here, $d_j(t)$ is equal to 1 plus the out-degree of j , and $W(t) = \sum_j \min\{d_j(t), A_2\}$ with the sum running over fertile vertices only. We refer to vertex $t + 1$ as a child of j . If after the first step, j has more than $A_1 - 1$ infertile children, one of them, chosen uniformly at random, becomes fertile. The process so defined is called a generalized preferential attachment process with fertility threshold A_1 and aging threshold A_2 .

Berger et al. show that in their model, the fraction of nodes with degree k converges to a limit q_k , which follows a power law for $k < A_2$, and decays exponentially above A_2 . Besides giving an explanation how the power law of the node degrees arises, their model may contribute to the design of computer networks, where the maximum degree is limited by some threshold.

2.4 Peer-to-Peer Overlay Networks

In this section, we survey peer-to-peer storage techniques for content distribution systems. We first characterize the notion of the peer-to-peer systems, and then describe several design issues, a classification of applications of peer-to-peer systems. We next review the overlay structures of peer-to-peer systems, including unstructured overlays and structured overlays. A comprehensive survey of peer-to-peer content distribution technologies can be found in [7].

2.4.1 Characteristics of Peer-to-Peer Systems

To date, there is no formal definition of peer-to-peer systems. A “pure” peer-to-peer network refers to a completely distributed system, where all nodes perform equivalent functionality and there is no central control over the entire network. However, such a description may not include current widely used peer-to-peer systems, such as Kazza [59], where there is a number of “supernodes” which serve as “coordinators” of a subset of

CHAPTER 2. MODELS OF LARGE-SCALE NETWORKS

tasks to some extent [7]. Currently the notion of the peer-to-peer systems covers from the completely decentralized distributed systems, such as Gnutella [42], to “partially centralized” systems, such as Kazaa. Androutsellis-Theotokis and Spinellis give the following definition [7]:

Peer-to-peer systems are distributed systems consisting of interconnected nodes able to self-organizing into network topologies with the purpose of sharing resources such as content, CPU cycles, storage and bandwidth, capable of adapting to failures and accommodating transient populations of nodes while maintaining acceptable connectivity and performance, without requiring the intermediation or support of a global centralized server or authority.

The notion of Peer-to-peer systems is somehow different from another large-scale distributed system, namely the Grid [24, 38]. The Grid systems aim to provide persistent, standard service infrastructures for geographically distributed computing and resource sharing [38], whereas the peer-to-peer systems emphasize more on the stability, fault tolerance, and self-adaptation [7]. On the other hand, as peer-to-peer systems become more sophisticated, and incorporate more complex applications, there will be a strong convergence between peer-to-peer and Grid systems [37].

The applications of peer-to-peer systems include: (1) Communication and Collaboration, such as Chat/IRC, Jabber [55], etc; (2) Distributed Computation, such as Seti@home [94], Genome@home [41], etc; (3) Internet Service Support, such as peer-to-peer multicast systems [89], Internet indirection infrastructures [96], etc.; (4) Database Systems, such as PIER [50], Piazza [47]; (5) Content Distribution, including most of current peer-to-peer systems, such as Napster [79], Gnutella [42], Kazaa [59], Oceanstore [63], Freenet [22], Chord [97], etc.

2.4.2 Design Issues in Peer-to-Peer Systems

Generally, the design issues of peer-to-peer systems include the following aspects.

- (1) **Scalability:** The capacity of the system should not be degraded drastically as the network size increases.
- (2) **Efficient resource locating:** There should be efficient mechanisms in the dynamic network to find the resources (e.g., files, computer power and storage space) fast without placing high demand on communications. Also, the system should be able to support complex queries, such as keyword search, range query, partial-matching searching, etc.
- (3) **Load Balancing:** The load balancing encompasses both storage balancing and query balancing. For storage balancing, the resource should be distributed over the network based on the node's capacity. For query balancing, the number of query messages via each node should be evenly distributed such that hot spots are avoided.
- (4) **Fault Tolerance:** The system should be robust enough to survive both node/link failures and malicious attacks.
- (5) **Anonymity:** The system should provide anonymity, such as the provider of the content, the node that stores the content, the identity of the content, etc.
- (6) **Security:** Due to the open and autonomous nature of peer-to-peer systems, there will be a number of research issues in secure storage, secure routing, access control, authentication, etc.

2.4.3 Network Structures

2.4.3.1 Classification

Current peer-to-peer structures can be classified into two categories: *structured* and *unstructured*. A peer-to-peer network is said to be *structured* if it is constructed deterministically according to specific rules rather than in *ad hoc* fashion. Otherwise, it is said to be an unstructured one.

In the structured peer-to-peer systems, the network topology is tightly controlled and the resources are stored at specific locations. The structured networks are normally implemented as Dynamic Hashed Tables (DHTs), where there is a mapping between the contents and the locations. Most structured peer-to-peer networks can support efficient and scalable exact-match queries, but cannot support other complex queries, such as partial-match or keyword queries, etc. Another disadvantage of the structured peer-to-peer networks is the overhead involved in maintaining the network topology, compared with the unstructured ones.

In the unstructured peer-to-peer networks, the network topology is set up in *ad hoc* fashion, and the placement of resource is independent of the network topology. Although the unstructured nature results in a lower maintenance cost, it also makes the locating of content more time consuming. Also, how to make the system scalable is a key issue of development. Although having these disadvantages, the unstructured peer-to-peer systems are widely used nowadays, such as Kazaa [59], Gnutella [42], etc. Probably it is because that they are easy to implement to support complex search, and also the servers of the network have not become a bottleneck so far.

In the next subsection, we will review some popular peer-to-peer systems developed so far, both in unstructured and structured categories.

2.4.3.2 Unstructured Peer-to-Peer Networks

Gnutella

Gnutella [42] is a purely decentralized peer-to-peer system, and each node services both client and server roles without central coordination of the activities in the network. The communications between nodes in Gnutella use the following types of messages.

- **Group Membership (PING and PONG) Messages:** Once a node joins the network, it broadcasts PING message to announce its presence. The nodes receiving the PING message forward the message to their neighbors and reply with a PONG message, which contains the information about its IP address, the port of the responding node and the size of shared files.
- **Search (QUERY and QUERY RESPONSE) Messages:** The QUERY message contains a search string and the specific requirement of the responding nodes. QUERY messages are broadcast in the same way of PING messages. The QUERY RESPONSE message is the reply message to QUERY, which contains the information necessary to download files.
- **File Transfer (GET and PUSH) Message:** The GET and PUSH messages are used for file downloading.

Each message in Gnutella has an ID number generated randomly to help back-propagation and prevent re-broadcasting. Also, each message is labelled with Time-to-Live (TTL) field to avoid swamping the network.

Freenet

Freenet [22] is a loosely “structured” system, in which each node can estimate the location of the file such that the blind broadcast is avoided. The main characteristics of Freenet include its emphasis on security, publisher anonymity, deniability, and data replication for availability and performance [7].

CHAPTER 2. MODELS OF LARGE-SCALE NETWORKS

In Freenet, each file is identified by a unique binary key. Each node maintains its local data store available to the network and a dynamic routing table consisting of files and their locations (nodes).

When a new node joins the network, it sends “Data Insert” message indicating the key and the actual data to some existing nodes in the network. Once other nodes receive the messages, they first check whether they have taken the key. If the key is not found, they forward the “Insert” message to a node having a closest key in the routing table. Each message is associated with a hop-to-live limit and the previous process is continued until the hops-to-live limit is reached. By doing so, newly published files are placed in more nodes than that contain files with similar keys.

During the file searching, when a node receives a request, it checks whether it stores the file, if not, it forwards the query message to its neighbor most likely to contain the file. To avoid a huge amount of traffic incurred, the messages are dropped when a certain number of hops, related to the hops-to-live, are passed, and a “data failed” message is passed back. If a node receives a “data failed” message from a downstream node, it then forwards the query message to the next best node possible to contain the file. If all the nodes in its routing table have been explored, it will send back a “data failed” message to its upstream node.

Once the requested file is found, a reply message will be sent back via the previous message-forwarded path. Also, the data will be cached in all the intermediate nodes along the path, which is useful for future requests.

Kazaa

Kazaa [59] is a partially “centralized” peer-to-peer system using the concept of *supernodes*. Some nodes with sufficient bandwidth and processing power are selected as supernodes, which function as servers for a small subset of normal nodes in the network. The jobs that supernodes serve include indexing the data shared by the normal nodes connected

CHAPTER 2. MODELS OF LARGE-SCALE NETWORKS

to them, caching the previous search results, sending the request messages for the sake of the normal nodes that they serve, etc. All queries are conducted among supernodes. Obviously, this reduces the query latency. Also, there is no single point of failure in the system. Once a supernode fails, one of the nodes connected to it will be selected as its substitute supernode. Thus, the network provides robustness and it is less vulnerable to malicious attacks. Another advantage of Kazaa is that it takes advantage of the network heterogeneity to construct the network topology.

2.4.3.3 Structured Peer-to-Peer Networks

Digital Tree-Based Schemes: Pastry and Tapestry

Plaxton [84] *et al.* propose a distributed routing protocol based on hypercubes for a static network with given collection of nodes. Plaxton's algorithm uses the *digit-fixing* technique to locate the shared resources on an overlay network in which each node only maintains a small-sized routing table. Pastry [91] and Tapestry [107] use Plaxton's scheme in a dynamic distributed environment. The difference between them is that Pastry uses *prefix-based* routing scheme, whereas Tapestry uses *suffix-based* scheme. Routing messages between any pair of nodes in Pastry or Tapestry can be finished in $O(\lg_B N)$ hops with the routing table $O(\lg_B N)$ for each node, where B denotes the number of bits for the sequence of digits. The number of bits per digit for both Tapestry and Pastry can be reconfigured but it remains fixed during run-time. Both Pastry and Tapestry can build the overlay topology using proximity neighbor selection. However, it is still unclear whether there is any better approach to achieve globally effective routing.

Ring-Based Schemes: Chord and Its Extensions

Chord [97] uses consistent hashing method to map n nodes to evenly distribute around a circle of identifiers. Each node x in Chord stores a pointer to its immediate *successor* (the closest node in clockwise direction along the circle), and a *finger table* of connections with node $x + 2^i$, where $i = 1, 2, \dots, \lg n - 1$, where n denotes the total number of nodes

CHAPTER 2. MODELS OF LARGE-SCALE NETWORKS

in the network. In Chord, a greedy algorithm is used to route query messages. The complexity of routing per query is bounded by $O(\lg n)$ hops. For fault tolerance, each node in Chord uses a successor list which stores the connections of next $\lg n$ successor nodes. The routing protocol in the original Chord in [97] is not optimal and has been improved by using bidirectional connections [40]. In [33], El-Ansary et al. generalize Chord to a P2P framework with k -ary search, but they only focus on the lookup operation, and do not consider node joining or leaving, node failures, and other implementation details. In [45, 71], a randomized version of Chord, called *Randomized-Chord*, is presented. In Randomized-Chord, node s is connected to a randomly chosen node in each interval $[2^{i-1}, 2^i)$, where $i = 1, 2, \dots, \lg n$. Koorde [57] is an extended DHT of Chord in that it embeds a de Bruijn graph over the Chord identifier circle. Koorde can be constructed with constant node degree and $O(\lg n)$ hops per query, or with $O(\lg n)$ node degree and $O(\lg n / \lg \lg n)$ hops per query. As a Chord-like network, Symphony [70] builds a network using the *small world* model from [61, 11]. In Symphony, each node has local links with its immediate neighbors and long distance links connected to randomly chosen nodes from a probability distribution function. The expected path length for a Symphony network with k links is $O(\frac{1}{k} \lg^2 n)$ hops. Simulations in [70] shows that Symphony is scalable, flexible, stable in a dynamic environment, and offers a small average latency with constant degree, but the analytical results for fault tolerance are not given. Like Chord, Viceroy [69] distributes nodes along a circle, and builds a constant-degree topology approximating a butterfly network, and offers $O(\lg n)$ routing latency. However, it is relatively complicated to implement Viceroy and fault tolerance is not addressed in [69].

Networks Based on d -dimensional Torus: CAN

CAN [87] divides a d -dimension torus space into zones *owned* by nodes, and resource keys are evenly hashed into the coordinate space. Each resource key is stored at the node that “owns” the zone. Using greedy routing, the query message is routed to the neighbor which

CHAPTER 2. MODELS OF LARGE-SCALE NETWORKS

	Lookup time	Routing table size	Peer join/leave overhead	Scalability	Hit ratio	Support complex search?	Load balancing	Fault tolerance
Plaxton	$O(\lg N)$	$O(\lg N)$	$\lg N$	Good	Good	No	Good	Fair
Chord	$O(\lg N)$	$O(\lg N)$	$(\lg N)^2$	Good	Good	No	Good	Good
CAN	$O(dN^{1/d})$	$O(d)$	$2d$	Good	Good	No	Good	-
Gnutella	$< TTL$	Constant	Constant	Fair	Fair	Yes	Fair	poor
Freenet	$< TTL$	Constant	Constant	Fair	Fair	Yes	Fair	Fair
Kazaa	$< TTL$	Constant	Constant	Fair	Fair	Yes	Fair	Fair

Table 2.3: Comparisons of different peer-to-peer structures.

is closer to the target key. Each node has $O(d)$ neighbors and query latency is $O(dn^{1/d})$. If d is chosen to be $\lg n$, each node connects with $O(\lg n)$ neighbors and a query takes $O(\lg n)$ hops. Some proximity routing schemes, such as *global network positioning* [81] and topologically-aware overlay construction [88], are used to build CAN overlay network. There are two shortcomings of these schemes: they need to fix some landmark machines and will create hot spots from a non-uniform distribution of nodes in the coordinate space.

2.4.4 Comparisons of Different Structures

Table 2.3 shows a comparison of different peer-to-peer structures. Most of the structured networks can achieve the $O(\lg N)$ lookup latency, with the routing table bounded by $O(\lg N)$, where N denotes the size of the network. CAN is an exception, where the lookup takes $O(dN^{1/d})$ hops, where d denotes the number of dimension employed. Also, most structured networks can locate rare content, but cannot support semantic search. While most unstructured systems support complex forms of search, they only support searching of popular items.

2.5 Relations between Social and Peer-to-Peer Networks

In this section, we will review recent study on the relations between social networks and peer-to-peer systems. Our survey is divided into three parts. First, we review the work on the extraction of social properties from real-world peer-to-peer systems. Second, we

look at the applications of social network principle to the design of peer-to-peer networks. Third, we examine the incentive mechanisms developed for peer-to-peer systems.

2.5.1 Observations of Social Behaviors from Real-World Peer-to-Peer Networks

It has been reported by empirical study that the degree distribution of early Gnutella networks obeys a power law distribution [1, 44, 90], that is, the fraction of nodes with k degree is proportional to $k^{-\alpha}$, where α is a network dependent constant. The power law distribution of node degree implies that a small number of nodes has a very large number of connections. As reported in [5, 16], the power law network displays a good robustness, that is, the network is robust to survive even each node or link fails randomly with high probability, but it is also vulnerable to adversarial attacks. The selective removal of a small number of nodes can disconnect the whole network. However, the empirical study by crawling Gnutella topology shows that, the recent (after 2001) Gnutella network has slightly deviated from the pure power distribution in that the formation of nodes with less than ten connections has changed to follow an almost constant distribution. The new Gnutella topology retains the robustness characteristics by becoming less dependent on nodes with high connections and it is therefore less vulnerable to adversarial attacks. As explained in [90], Gnutella users normally have less bandwidth available compared with that of the normal Internet users. Moreover, the users intend to set up as many connections as possible because of their perception and experience that more connections will lead to better query results.

In [56], the Gnutella network is also shown to exhibit (moderate) small-world characteristics, that is, both small diameter and high clustering coefficient compared with traditional random graphs. Iamnitchi et al. [52] explore how the small-world characteristics is generated in Gnutella. They claim that, although it is possible that the small-world network topology reflects the social network formed by the Gnutella users, the small-world

CHAPTER 2. MODELS OF LARGE-SCALE NETWORKS

phenomenon is more likely to have been influenced by the technical reasons rather than the social reasons. For instance, in the network exploration protocol of Gnutella, a node sends a PING message to all its neighbors, and then each neighbor forwards the message to its own neighbors, and so on. A PONG message is sent back along the same path so that the neighbors know each other. Such a mechanism will have led to high clustering coefficient [52].

In [53], Iamnitchi et al. propose the concept of *data sharing graph* to capture common user interests in data. In the data sharing graph, users are represented by nodes, and any two users are connected if they share some similar interest in data within some interval. The authors in [53] empirically report that all the data sharing graphs of three communities: the collaboration network among the physicists, the Web, and the Kazaa peer-to-peer network, all display the small-world pattern, that is, there are both large clustering coefficient and short path exist in general in the data sharing graphs. The small-world properties in the peer-to-peer network imply that flooding with small radius should be able to span the most of the graph [53]. In [53] and [66], the degree distribution of Kazaa network is also found to follow the power law distribution.

2.5.2 Applications of Social Studies to the Peer-to-Peer Network Design

Social studies have revealed that the small-world graphs have small diameters and large clustering coefficient. The findings have been applied to the design of peer-to-peer networks supporting content search.

In [106], Zhang et al. propose an enhanced-clustering cache replacement scheme based on the small-world models and Kleinberg's theoretical results [61] to improve Freenet's performance. In [70], Manku et al. propose Symphony, Dynamic Hash Table (DHT) that applies Kleinberg's small-world model to set up connections between nodes. In Symphony, nodes' addresses and file contents are hashed into nodes uniformly distributed along a ring,

CHAPTER 2. MODELS OF LARGE-SCALE NETWORKS

and each node stores the indexes of the resources that are hashed to lie between itself and its predecessor. Besides local links connecting to the predecessor and the successor, each node has a long-range link, which is set up in the same way as in Kleinberg's small-world model.

In [95], Sripanidkulchai et al. improve the current flooding approach in unstructured peer-to-peer systems by exploiting interest-based shortcuts between peers. In their scheme, each node establishes several shortcuts according to query history. Intuitively, such a mechanism will enable the graph to evolve into a small-world network eventually. To some extent, their experimental results have verified this intuition.

In order to support both partial-match queries and efficient location of rare items, Cohen et al. [23] propose an associative overlay network built on current unstructured peer-to-peer systems, such as Gnutella. Their approach is based on the existence of association in the society so that people normally steer the search process to peers that are more likely to have the answers.

In [67], Li et al. present a *Semantic Small World* (SSW), a structured overlay network, to support efficient semantic search. In SSW, nodes are clustered and organized based on the semantics of data that they store locally. Then the clusters are self-organized into a small-world network, which displays low search latency and low maintenance cost.

2.5.3 Incentive Mechanisms for Peer-to-Peer Networks

Another interesting aspect of some peer-to-peer networks is that the users in these decentralized systems participate in the network voluntarily according to their own interest. It is necessary to employ some incentive mechanisms to effectively facilitate cooperative behaviors among the users. Without such incentive mechanisms, the systems may have poor performance for content sharing. For example, in current peer-to-peer systems, such as Kazaa, there is an uncooperative behavior called "free riding", which means that users only download the files without sharing their own files.

The implementation of the incentive mechanism in peer-to-peer systems is much more difficult than in centralized systems (such as eBay), because it is hard to know past behaviors of each peer, and to predict their future performance. The current solutions can be classified into two categories: (1) Trust-based mechanism in which whether two peers engage in a transaction depend on whether they trust each other; and (2) Trade-based mechanism in which the peer or party provides some service to one another based on the explicit remunerability. Here we would not go into more details of these schemes. The reader is referred to [7] for more comprehensive survey.

2.6 Prospective Research

Although there have been a great number of models for small-world and power law phenomena, and structures for peer-to-peer computing, there are still some rooms of improvements for the existing schemes. We choose the following topics to be our research focuses.

- (1) *More effective models for large-scale distributed networks.* Although the models proposed for large-scale distributed networks exhibit the desirable small diameter or power law distribution of node degree, they are far from approximating the real-world complex networks. For example, both Kleinberg's model for small-world phenomenon [61] and the BA model for the power law distribution of node degree [10] require the global information of the current network status, which is an impractical assumption for modelling the real-world complex networks. Thus, our research aims to propose a more effective model that better approximate the real-world networks.
- (2) *More efficient decentralized routing algorithms for Kleinberg's small-world model.* All existing decentralized routing algorithms [8, 65, 39, 72] can only achieve $O((\log n)^{1+\epsilon})$ expected number of hops, where $\epsilon > 0$ denotes a constant. Martel and Nguyen [72] have found that the expected diameter of Kleinberg's small-world networks is $\Theta(\log n)$. Thus a question arises naturally: Can we improve the

CHAPTER 2. MODELS OF LARGE-SCALE NETWORKS

routing algorithms to match the diameter of the networks while keeping the amount of information stored on each node as small as possible?

Chapter 3

Evolution of Scale-Free Networks with Partial Information

As we reviewed in Chapter 2, most known models for scale-free networks require each node to know about all the other nodes in the network, which is rather unrealistic for large-scale distributed systems. In this chapter, we will propose a model where the nodes use less information. We will first show theoretically that the power law of degree distribution still holds true even if each new joining node knows only a logarithmic number of the existing nodes. We then experimentally verify it by using simulations.

The remainder of this chapter is organized as follows. Section 3.1 further describes the problem issues and our approach. Section 3.2 presents our model and some notations. Section 3.3 lists our main results. In Section 3.4, we theoretically prove our results. In Section 3.5, we experimentally verify these results. Section 3.6 describes some related work. Section 3.7 summarizes our work in this chapter.

3.1 Introduction

Recently major advances have been witnessed in modelling real-world large-scale networks such as the Internet and the Web [2, 10, 13, 18, 20, 21, 25, 30, 34, 64, 78, 43, 100]. A mathematical model is useful for studying the properties of the real-world complex networks such as the node degree distribution [18], connectivity [77], maximum degree [25],

CHAPTER 3. EVOLUTION OF SCALE-FREE NETWORKS WITH PARTIAL INFORMATION

diameter [17], etc. Knowledge concerning the properties of these networks and the evolution mechanisms of them may find applications in many contexts, such as the construction of the application overlays [88], Internet attacks [16], and virus spreading [54].

Recall that both the Internet and the Web exhibit a *scale-free* (i.e., size-independent) property: The proportion P_k of nodes with degree k follows a power law distribution: $P_k \sim k^{-r}$, where $r \leq 3$. To model such scale-free networks, Barabási and Albert [10] proposed the first model (the BA model) based on a preferential attachment (PA) rule, i.e., an existing node with higher node degree is more likely to get a link from the new node. Motivated by the fact that a new web node is often generated by copying an old one and changing some of its old links, Kumar et al. [64] independently presented a so-called *copying model*. Fabrikant, Koutsoupias, and Papadimitriou (FKP) [34] proposed a “Highly Optimized Tolerance” (HOT) framework to model the Internet, where the connection of the new node is constructed according to an adjustable neighborhood. A variation of the BA model, in which each node has an “initial attractiveness”, was introduced independently by two different groups, Dorogovtsev et al. [30] and Drinea et al. [31]. A precise and rigorous analysis of the model with initial attractiveness was given by Buckley and Osthus [21]. Cooper and Frieze [25] presented a general model which considers variable number of edges or nodes generated at each time step, a mixture of uniform and PA selection, etc.

Although these scale-free models are consistent with the power law observation of large-scale networks, unfortunately, most of existing models assume that a newly added node knows about all of the existing nodes, such as their node degrees [10], the Euclidean distance [34], or the hop distance to the network center [13], etc. This is an unrealistic assumption with large-scale networks such as the Internet or the Web, as it requires a node to access and process an extremely large amount of information. We will, instead, assume that each node has access to only a small subset of logarithmic size of all the existing nodes.

As with [10, 17], we also allow only one node to be added into the network at each time step. However, in our model, each node uniformly and randomly chooses a subset of existing nodes, and then connects to some nodes in this sample set according to the PA rule. Our results show that the power law of degree distribution still holds true even if each new joining node knows only a logarithmic number of the existing nodes. Compared to the existing models, our construction of scale-free networks based on partial information seems to better approximate the evolution of real-world complex networks.

Our results may also be applied to the constructions of large-scale distributed systems such as peer-to-peer networks, where the global information is generally unavailable for any individual node in the network.

3.2 The Model and Notations

Let m denote an integer constant. Our construction is described as follows.

Step 1: Start with G_1^m , the graph with only one single node denoted by v_1 with m self-loops.

Step t : A new node denoted by v_t is added to the graph G_{t-1}^m to form G_t^m . This new node sends m edges to the existing nodes in G_{t-1}^m according to the following rule: First randomly and independently choose S_{t-1} nodes from G_{t-1}^m to form the sample set T_{t-1} . Then node v_t sends m edges to the nodes in the sample set T_{t-1} according to the preferential attachment rule, i.e., the probability that v_t is connected to a node $u \in T_{t-1}$ is

$$Pr[v_t \rightarrow u] = \frac{deg_{t-1}(u)}{\sum_{i \in T_{t-1}} deg_{t-1}(i)},$$

where $deg_{t-1}(x)$ denotes the degree of node x in the graph G_{t-1}^m .

We will show that to ensure the power law, it suffices to choose $S_t = \beta \lg t$, where β denotes a constant to be specified later. During the initial steps, it is possible that $S_t \leq t$. At the initial stage, we choose all the existing nodes as the sample set. Such initial

CHAPTER 3. EVOLUTION OF SCALE-FREE NETWORKS WITH PARTIAL INFORMATION

choices will not affect our asymptotic results. We assume that all edges in the graph are undirected. The directed variant of our model can be easily obtained by applying a similar method as in [25].

Below are the notations that will be used in our analysis.

$\deg_t(x)$: The degree of node x in the graph G_t^m ;

T_t : The sample set chosen at the time step t ;

S_t : The size of the sample set T_t , i.e., $S_t = |T_t|$;

D_t : The sum of the node degrees of all nodes in the sample set T_t , i.e., $D_t = \sum_{x \in T_t} \deg_t(x)$;

$d_t(i)$: The number of nodes with degree i in the graph G_t^m ;

$d_{t,s}(i)$: The number of nodes with degree i in the sample set T_t ;

$\bar{d}_t(i)$: The expectation of $d_t(i)$;

$\bar{d}_{t,s}(i)$: The expectation of $d_{t,s}(i)$;

M_t : The maximum degree of the graph G_t^m ;

$\text{Diam}(G_t^m)$: The diameter of the graph G_t^m ;

n : The size of the final graph G_n^m .

Throughout the chapter, we will identify the node v_x with the integer number x to simplify the notation.

3.3 Our Main Results

Our main results are stated as follows:

Result 1 (Node Degree Distribution): Let m and β denote sufficiently large constants, and let $S_t = \beta \lg t$ denote the size of the sample set in Step t . Let $d_n(i)$ denote the number of nodes with degree i in G_n^m . Then *with high probability (whp)*

$$d_n(i) = \frac{cn}{i(i+1)(i+2)} + O(n^\theta),$$

where θ and c denote constants, and $0 < \theta < 1$.

Since each node has a degree of at least m , it is trivial to calculate $d_t(i)$ for $i \leq m$. In the subsequent analysis, when we talk about $d_t(i)$, we assume that $i > m$.

Result 2 (Maximum Degree): Let M_n denote the maximum degree of the graph G_n^m . Then for any constant $\epsilon \in (0, \frac{1}{2})$, *with high probability (whp)*

$$C_2 n^{\frac{1}{2}-\epsilon} \leq M_n \leq C_1 n^{\frac{1}{2}+\epsilon},$$

where C_1 and C_2 denote constants.

Result 3 (Network Diameter): Let $\text{Diam}(G_n^m)$ denote the diameter of the graph G_n^m . If $m \geq C_3 \lg n$ for a sufficiently large constant C_3 , then *with high probability (whp)*

$$\text{Diam}(G_n^m) \leq 2 \lg n.$$

Although we will only show that $\text{Diam}(G_n^m) = O(\lg n)$ for the case $m = \Omega(\lg n)$, we conjecture that $\text{Diam}(G_n^m) = O(\lg n)$ also holds for $m = \Theta(1)$.

The proof of **Result 1** is relatively more challenging and its proof is given in Section 3.4.1. **Result 2** (Theorem 3.4) is proven in Section 3.4.2, whereas **Result 3** (Theorem 3.5) is proven in Section 3.4.3.

3.4 Mathematical Analysis

3.4.1 The Degree Distribution

The following is a roadmap for the proof of **Result 1**. Firstly, Section 3.4.1.1 gives the expectation of D_t and shows that D_t concentrates around its expectation by applying an extension of the martingale method [73]. Secondly, based on the concentration result of D_t , a recurrence relation of $\bar{d}_t(i)$ is given in Section 3.4.1.2. We then inductively show that its solution follows a power law. In Section 3.4.1.3, we argue that $d_t(i)$ concentrates around its expectation by applying a similar analysis of the concentration of D_t .

3.4.1.1 Concentration of D_t

We first analyze the expectation of D_t .

Since the sample nodes in T_t are selected randomly and independently from the current nodes in G_t^m , we can easily obtain the following lemma.

Lemma 3.1 $E[d_{t,s}(i)] = \frac{S_t}{t} E[d_t(i)]$.

Based on the above lemma, the expectation of D_t can be easily obtained.

Lemma 3.2 $E[D_t] = 2mS_t$.

Proof:

$$\begin{aligned} E[D_t] &= E\left[\sum_{i>0} d_{t,s}(i)\right] = \sum_{i>0} E[d_{t,s}(i)] = \sum_{i>0} \frac{S_t}{t} E[d_t(i)] \quad (\text{by Lemma 3.1}) \\ &= \frac{S_t}{t} \sum_{i>0} E[d_t(i)] = \frac{S_t}{t} E\left[\sum_{i>0} d_t(i)\right] = \frac{S_t}{t} 2mt = 2mS_t. \end{aligned}$$

■

Our analysis of the concentration result is mainly based on the following probabilistic tool, which is an extension of the martingale method [6].

CHAPTER 3. EVOLUTION OF SCALE-FREE NETWORKS WITH PARTIAL INFORMATION

Lemma 3.3 (Martingale extension [73]) *Let $X = (X_1, \dots, X_n)$ be a family of random variables with X_k taking value in a set A_k , and let f be a bounded real-valued function defined on $\Pi A_k = A_1 \times A_2 \times \dots \times A_n$. Let $x_i \in A_i$ for each $i = 1, \dots, k-1$. For $x \in A_k$, let $g_k(x) = E[f(X)|X_k = x] - E[f(X)]$. Let $r_k = \sup\{|g_k(x) - g_k(y)| : x, y \in A_k\}$, and $R^2(X) = \sum_{k=1}^n r_k^2$. Let $\hat{r}^2 = \sup\{R^2(X) \text{ for all } X \in \Pi A_k\}$. Then*

$$\Pr[|f(X) - E[f(X)]| \geq c] \leq 2 \exp(-2c^2/\hat{r}^2),$$

where $c > 0$

The following lemma gives a lower bound of D_t which is useful for our subsequent analysis.

Lemma 3.4 *Let m and β denote sufficiently large constants, and let $S_t = \beta \lg t$ as defined earlier and let $n_0 = n^{5/6}$. Then there exists a constant δ where $0 < \delta < 1$, such that*

$$\Pr[D_t \geq (1 + \delta)mS_t] > 1 - \frac{1}{n^2},$$

for all $t > n_0$.

Proof: Let D_t^i denote the degree of the i th selected node of the sample set T_t , where $1 \leq i \leq S_t$. Note that $D_t = \sum_{i=1}^{S_t} D_t^i$. We give the expectation of D_t^i as follows.

$$E[D_t^i] = \sum_{j>0} d_t(j) \cdot \frac{1}{t} \cdot j = \frac{1}{t} \sum_{j>0} d_t(j) \cdot j = \frac{1}{t} \cdot 2mt = 2m.$$

The following lemma shows that $\text{Var}[D_t^i] \leq E[D_t^i]$.

Lemma 3.5 $\text{Var}[D_t^i] \leq E[D_t^i]$.

Proof of Lemma 3.5: As in [18, 17], we regard one edge as two “half-ward” edges. Let $Z_{k,j}$ be the indicator that the j th half edge connected to a node with degree k is selected,

CHAPTER 3. EVOLUTION OF SCALE-FREE NETWORKS WITH PARTIAL INFORMATION

where $1 \leq j \leq k$. Then we have $E[Z_{k,j_1}] = E[Z_{k,j_2}] = \frac{d_t(k)}{t}$ for $1 \leq j_1, j_2 \leq k$ and $j_1 \neq j_2$, and $D_t^i = \sum_{k>0} \sum_{j=1}^k Z_{k,j}$. We now estimate the second moment of D_t^i :

Since $E[(Z_{k,j})^2] = E[Z_{k,j}]$, we have

$$\begin{aligned} E[(D_t^i)^2] &= E\left[\sum_{k>0} \sum_{j=1}^k (Z_{k,j})^2 + \sum_{k_1, k_2 > 0; k_1 \neq k_2} \sum_{1 \leq j_1 \leq k_1; 1 \leq j_2 \leq k_2} Z_{k_1, j_1} \cdot Z_{k_2, j_2}\right] \\ &= E\left[\sum_{k>0} \sum_{j=1}^k Z_{k,j} + \sum_{k_1, k_2 > 0; k_1 \neq k_2} \sum_{1 \leq j_1 \leq k_1; 1 \leq j_2 \leq k_2} Z_{k_1, j_1} \cdot Z_{k_2, j_2}\right] \end{aligned}$$

Since $E[Z_{k,j_1}] = E[Z_{k,j_2}] = \frac{d_t(k)}{t}$ for $1 \leq j_1, j_2 \leq k$ and $j_1 \neq j_2$, we have

$$E\left[\sum_{j=1}^k Z_{k,j}\right] = k \frac{d_t(k)}{t}.$$

Thus,

$$\begin{aligned} E[(D_t^i)^2] &= \sum_{k>0} E\left[\sum_{j=1}^k Z_{k,j}\right] + \sum_{k_1, k_2 > 0; k_1 \neq k_2} E\left[\sum_{1 \leq j_1 \leq k_1; 1 \leq j_2 \leq k_2} Z_{k_1, j_1} \cdot Z_{k_2, j_2}\right] \\ &= \sum_{k>0} k \frac{d_t(k)}{t} + \sum_{k_1, k_2 > 0; k_1 \neq k_2} \left(k_1 \frac{d_t(k_1)}{t}\right) \left(k_2 \frac{d_t(k_2)}{t}\right) \\ &\leq E[D_t^i] + \sum_{k_1, k_2 > 0; k_1 \neq k_2} \left(k_1 \frac{d_t(k_1)}{t}\right) \left(k_2 \frac{d_t(k_2)}{t}\right) \\ &= E[D_t^i] + (E[D_t^i])^2. \end{aligned}$$

So $\text{Var}[D_t^i] = E[(D_t^i)^2] - (E[D_t^i])^2 \leq E[D_t^i]$. ■

By Lemma 3.5, for a constant $0 < \alpha < 1$, we have

$$\begin{aligned} \Pr[D_t^i \leq (1 + \alpha)m] &\leq \Pr[D_t^i \leq (1 + \alpha)m] + \Pr[D_t^i \geq (3 - \alpha)m] \\ &= \Pr[D_t^i \leq 2m - (1 - \alpha)m] + \Pr[D_t^i \geq 2m + (1 - \alpha)m] \\ &= \Pr[|D_t^i - 2m| \geq (1 - \alpha)m] \\ &\leq \frac{\text{Var}[D_t^i]}{((1 - \alpha)m)^2} \quad (\text{by Chebyshev's inequality}) \\ &\leq \frac{E[D_t^i]}{((1 - \alpha)m)^2} = \frac{2m}{((1 - \alpha)m)^2} = \frac{2}{m(1 - \alpha)^2} \end{aligned}$$

CHAPTER 3. EVOLUTION OF SCALE-FREE NETWORKS WITH PARTIAL INFORMATION

Thus, $\Pr[D_t^i \geq (1 + \alpha)m] \geq 1 - \frac{2}{(1 - \alpha)^2 m}$, which is greater than a positive constant when m is sufficiently large.

Let I_i denote the indicator variable of the event that $D_t^i \geq (1 + \alpha)m$, $1 \leq i \leq S_t$. Since all I_i are independent from each other, $\sum_{i=1}^{S_t} I_i$ stochastically dominates the binomial distribution $\text{Bi}(S_t, 1 - \frac{2}{(1 - \alpha)^2 m})$. By Chernoff's bound, for $0 < \epsilon < 1$, we have

$$\Pr \left[\sum_{i=1}^{S_t} I_i \leq (1 - \epsilon) \left(1 - \frac{2}{(1 - \alpha)^2 m}\right) S_t \right] \leq \exp \left(- \frac{\epsilon^2}{2} \left(1 - \frac{2}{(1 - \alpha)^2 m}\right) S_t \right)$$

Thus

$$\Pr \left[\sum_{i=1}^{S_t} I_i \geq (1 - \epsilon) \left(1 - \frac{2}{(1 - \alpha)^2 m}\right) S_t \right] \geq 1 - \exp \left(- \frac{\epsilon^2}{2} \left(1 - \frac{2}{(1 - \alpha)^2 m}\right) S_t \right)$$

So

$$\begin{aligned} \Pr \left[\sum_{i=1}^{S_t} D_t^i \geq (1 - \epsilon) \left(1 - \frac{2}{(1 - \alpha)^2 m}\right) S_t (1 + \alpha)m \right] &\geq \Pr \left[\sum_{i=1}^{S_t} I_i \geq (1 - \epsilon) \left(1 - \frac{2}{(1 - \alpha)^2 m}\right) S_t \right] \\ &\geq 1 - \exp \left(- \frac{\epsilon^2}{2} \left(1 - \frac{2}{(1 - \alpha)^2 m}\right) S_t \right) \end{aligned}$$

Choosing sufficiently large m and suitable constant $0 < \alpha < 1$, we have

$$\Pr \left[\sum_{i=1}^{S_t} D_t^i \geq (1 + \delta)m S_t \right] \geq 1 - \exp \left(- \frac{\epsilon^2}{2} \left(1 - \frac{2}{(1 - \alpha)^2 m}\right) S_t \right),$$

for a constant $0 < \delta < 1$.

When $t > n_0 = n^{5/6}$, we have $S_t = \beta \log t > \beta \log n^{5/6} = \frac{5}{6} \beta \log n$. So

$$\Pr \left[\sum_{i=1}^{S_t} D_t^i \geq (1 + \delta)m S_t \right] \geq 1 - \exp \left(- \frac{\epsilon^2}{2} \left(1 - \frac{2}{(1 - \alpha)^2 m}\right) \frac{5}{6} \beta \log n \right),$$

As long as we choose sufficiently large constants m and β , we can obtain $\Pr \left[\sum_{i=1}^{S_t} D_t^i \geq (1 + \delta)m S_t \right] \geq 1 - \frac{1}{n^2}$ for a constant $0 < \delta < 1$. Therefore, the proof of Lemma 3.4 is completed. ■

CHAPTER 3. EVOLUTION OF SCALE-FREE NETWORKS WITH PARTIAL INFORMATION

Let N_i denote the set of neighbors of node i when it first enters the network at step i . Then N_i is a tuple of nodes $\vec{x} = (x_1, \dots, x_m) \in \{1, \dots, i-1\}^m$. Let $g_{\tau,t}(\vec{x}) = E[D_t | N_1, \dots, N_{\tau-1}, N_\tau = \vec{x}]$, where the sequence of $N_1, \dots, N_{\tau-1}$ is fixed and $\vec{x} \in \{1, \dots, \tau-1\}^m$, $1 \leq \tau \leq t$. Let $r_{\tau,t} = \sup \{|g_{\tau,t}(\vec{x}) - g_{\tau,t}(\vec{y})| : \vec{x}, \vec{y} \in \{1, \dots, \tau-1\}^m\}$. In order to bound $r_{\tau,t}$ and apply the martingale extension in Lemma 3.3 to analyze the concentration of D_t , we introduce the following node-edge marking rule:

As in [18, 17], we regard one edge as two “half-ward” directed edges. At a time step τ , we initially mark the nodes in the set N_τ as “ τ -influenced nodes”, and mark all the half edges connected to N_τ as “ τ -influenced half edges”. During the next time step $\tau+1$, if an outgoing half edge sent by the new node $\tau+1$ is connected to a τ -influenced node, it is also marked as a τ -influenced edge. Applying the same rule for the constructions of the sequence of graphs $G_{\tau+2}^m, \dots, G_t^m$, then the value of $E[D_t | N_1, \dots, N_{\tau-1}, N_\tau = \vec{x}] - E[D_t | N_1, \dots, N_{\tau-1}, N_\tau = \vec{y}]$ is upper bounded by the number of all τ -influenced edges that are attached to the sample set T_t . Let Δ_σ denote the expected number of τ -influenced edges in the graph G_σ^m , then we have $r_{\tau,t} \leq \frac{S_t}{t} \Delta_t$.

Based on the above observations, we have the following lemma.

Lemma 3.6 *Define $r_{\tau,t}$ and Δ_τ as above. Let m and β denote sufficiently large constants, and let $S_t = \beta \lg t$. Let δ denote a constant such that $0 < \delta < 1$. Then,*

$$r_{\tau,t} \leq \begin{cases} \frac{S_t}{t} \Delta_\tau \frac{n_0}{\tau} \left(\frac{t}{n_0}\right)^{\frac{1}{1+\delta}}, & \text{when } 1 \leq \tau \leq n_0; \\ \frac{S_t}{t} \Delta_\tau \left(\frac{t}{\tau}\right)^{\frac{1}{1+\delta}}, & \text{when } n_0 < \tau \leq t. \end{cases}$$

where $n_0 = n^{5/6}$.

Proof: Recall that Δ_σ is the expected number of τ -influenced edges in the graph G_σ^m . Let Y_σ denote the expected number of new τ -influenced edges generated from G_σ^m during step $\sigma+1$. Then by linearity of expectation, we have $\Delta_{\sigma+1} = \Delta_\sigma + Y_\sigma$.

CHAPTER 3. EVOLUTION OF SCALE-FREE NETWORKS WITH PARTIAL INFORMATION

Let e_i^σ denote the number of nodes with i τ -influenced edges among G_σ^m , then $\Delta_\sigma = \sum_{i>0} i \cdot e_i^\sigma$. So

$$Y_\sigma \leq \sum_{i>0} \frac{m \cdot e_i^\sigma \cdot \frac{S_\sigma}{\sigma} \cdot i}{D_\sigma} = \frac{mS_\sigma}{\sigma D_\sigma} \sum_{i>0} e_i^\sigma \cdot i = \frac{mS_\sigma \Delta_\sigma}{\sigma D_\sigma}.$$

We bound Y_σ according to two different cases.

Case 1 $n_0 < \sigma < t$: Let $\mathcal{E}_\sigma$ denote the event that $D_\sigma \geq (1 + \delta)mS_\sigma$. From Lemma 3.4, we have $\Pr[\mathcal{E}_\sigma] \geq 1 - \frac{1}{n^2}$. Let $\mathcal{E} = \bigcap_{\sigma=\tau}^t \mathcal{E}_\sigma$, then the event $\mathcal{E}$ occurs with probability at least $1 - \frac{1}{n}$. Thus we can assume that $D_\sigma \geq (1 + \delta)mS_\sigma$ for all $n_0 < \sigma \leq t$ in the following. Thus,

$$Y_\sigma \leq \frac{mS_\sigma \Delta_\sigma}{\sigma D_\sigma} \leq \frac{\Delta_\sigma}{(1 + \delta)\sigma}$$

Case 2 $\tau < \sigma \leq n_0$: It is obvious that $D_\sigma \geq mS_\sigma$. So we have

$$Y_\sigma \leq \frac{mS_\sigma \Delta_\sigma}{\sigma D_\sigma} \leq \frac{\Delta_\sigma}{\sigma}$$

Combining the above two cases, when $1 \leq \tau \leq n_0$, we have

$$\begin{aligned} \Delta_t &\leq \Delta_\tau \prod_{\sigma=\tau+1}^{n_0} \frac{\sigma+1}{\sigma} \prod_{\sigma=n_0+1}^t \left(1 + \frac{1}{(1+\delta)\sigma}\right) \\ &\leq \Delta_\tau \frac{n_0}{\tau} \left(\frac{t}{n_0}\right)^{\frac{1}{1+\delta}} \end{aligned}$$

By using the fact that $1 + ax \leq (1 + x^a)$ for $x > -1$ and $a \geq 1$, we have $1 + \frac{1}{(1+\delta)\sigma} \leq (1 + \frac{1}{\sigma})^{1/(1+\delta)}$. Thus, we have

$$\begin{aligned} \Delta_t &\leq \Delta_\tau \prod_{\sigma=\tau+1}^{n_0} \frac{\sigma+1}{\sigma} \prod_{\sigma=n_0+1}^t \left(1 + \frac{1}{\sigma}\right)^{\frac{1}{1+\delta}} \\ &\leq \Delta_\tau \cdot \left(\frac{\tau+2}{\tau+1} \times \frac{\tau+3}{\tau+2} \times \cdots \times \frac{n_0+1}{n_0}\right) \cdot \left(\frac{n_0+2}{n_0+1} \times \frac{n_0+3}{n_0+2} \times \cdots \times \frac{t+1}{t}\right)^{\frac{1}{1+\delta}} \\ &\leq \Delta_\tau \frac{n_0}{\tau} \left(\frac{t}{n_0}\right)^{\frac{1}{1+\delta}} \end{aligned}$$

CHAPTER 3. EVOLUTION OF SCALE-FREE NETWORKS WITH PARTIAL INFORMATION

When $n_0 < \tau < t$, only case 1 applies, so we have

$$\begin{aligned} \Delta_t &\leq \Delta_\tau \prod_{\sigma=\tau+1}^t \left(1 + \frac{1}{(1+\delta)\sigma}\right) \\ &\leq \Delta_\tau \prod_{\sigma=\tau+1}^t \left(1 + \frac{1}{\sigma}\right)^{\frac{1}{1+\delta}} \quad \left(\text{since } 1 + \frac{1}{(1+\delta)\sigma} \leq \left(1 + \frac{1}{\sigma}\right)^{\frac{1}{1+\delta}}\right) \\ &\leq \Delta_\tau \left(\frac{\tau+2}{\tau+1} \times \frac{\tau+3}{\tau+2} \times \cdots \times \frac{t+1}{t}\right)^{\frac{1}{1+\delta}} \\ &\leq \Delta_\tau \left(\frac{t}{\tau}\right)^{\frac{1}{1+\delta}} \end{aligned}$$

Since $r_{\tau,t} \leq \frac{S_t}{t} \Delta_t$, we have

$$r_{\tau,t} \leq \begin{cases} \frac{S_t}{t} \Delta_\tau \frac{n_0}{\tau} \left(\frac{t}{n_0}\right)^{\frac{1}{1+\delta}}, & \text{when } 1 \leq \tau \leq n_0; \\ \frac{S_t}{t} \Delta_\tau \left(\frac{t}{\tau}\right)^{\frac{1}{1+\delta}}, & \text{when } n_0 < \tau \leq t. \end{cases}$$

■

Theorem 3.1 Let m and β denote sufficiently large constants, and let $S_t = \beta \lg t$ as before. Let $n_1 = n^{\frac{11}{12}}$. Then there exists a constant $0 < \varphi < 1$ such that

$$\Pr[|D_t - 2mS_t| \geq mS_t t^{\varphi-1}] \leq \frac{1}{n^2},$$

for all $t > n_1$.

Proof: Let $R_t^2 = \sum_{\tau=1}^t (r_{\tau,t})^2$. From Lemma 3.6, we have

$$\begin{aligned} R_t^2 &= \sum_{\tau=1}^t (r_{\tau,t})^2 \\ &\leq \sum_{\tau=1}^{n_0} \left(\frac{S_t}{t} \Delta_\tau \frac{n_0}{\tau} \left(\frac{t}{n_0}\right)^{\frac{1}{1+\delta}}\right)^2 + \sum_{\tau=n_0+1}^t \left(\frac{S_t}{t} \Delta_\tau \left(\frac{t}{\tau}\right)^{\frac{1}{1+\delta}}\right)^2 \\ &= \left(\frac{S_t}{t} \Delta_\tau t^{\frac{1}{1+\delta}}\right)^2 (n_0^{2-\frac{2}{1+\delta}} \sum_{\tau=1}^{n_0} \frac{1}{\tau^2} + \sum_{\tau=n_0+1}^t \frac{1}{\tau^2}) \\ &= O\left(\left(\frac{S_t}{t} \Delta_\tau t^{\frac{1}{1+\delta}} n_0^{1-\frac{1}{1+\delta}}\right)^2\right) \end{aligned}$$

Since node τ affects at most $2m$ degrees at the time step τ , $\Delta_\tau \leq 2m$. Hence, $R_t^2 = O\left(\left(\frac{S_t}{t} \Delta_\tau t^{\frac{1}{1+\delta}} n_0^{1-\frac{1}{1+\delta}}\right)^2\right) = O\left(\left(\frac{S_t}{t} m t^{\frac{1}{1+\delta}} n_0^{1-\frac{1}{1+\delta}}\right)^2\right).$

CHAPTER 3. EVOLUTION OF SCALE-FREE NETWORKS WITH PARTIAL INFORMATION

So $\hat{r}^2 = \sup\{R_t^2\} = O((\frac{S_t}{t} m t^{\frac{1}{1+\delta}} n_0^{1-\frac{1}{1+\delta}})^2)$. By Lemma 3.3, we have

$$\Pr[|D_t - 2mS_t| \geq \frac{mS_t}{t} t^{\frac{1}{1+\delta}} n_0^{1-\frac{1}{1+\delta}} \lg n] = \exp(-\Omega(\lg^2 n))$$

Since $n_1 = n^{\frac{11}{12}} > n_0 = n^{\frac{5}{6}}$, there exists a constant $0 < \varphi < 1$ such that

$$\Pr[|D_t - 2mS_t| \geq mS_t t^{\varphi-1}] \leq \frac{1}{n^2},$$

for all $t > n_1$ ■

3.4.1.2 Power Law Distribution of $\bar{d}_n(i)$

Theorem 3.2 *Let m and β denote sufficiently large constants, and let $S_t = \beta \lg t$. Then there exists a constant $0 < \theta < 1$ such that **whp***

$$\bar{d}_n(i) = \frac{cn}{i(i+1)(i+2)} + O(n^\theta),$$

for a constant c .

Proof: By construction of our model (cf. Section 3.2), we have the following relation:

$$E[d_{t+1}(i)|G_t^m] = d_t(i) + m d_{t,s}(i-1) \frac{i-1}{D_t} - m d_{t,s}(i) \frac{i}{D_t}.$$

Taking the expectation on both sides, we have

$$E[d_{t+1}(i)] = E[d_t(i)] + m(i-1)E\left[\frac{d_{t,s}(i-1)}{D_t}\right] - mE\left[\frac{d_{t,s}(i)}{D_t}\right]. \quad (3.1)$$

Let $\mathcal{F}$ denote the event that $|D_t - 2mS_t| < mS_t t^{\varphi-1}$, then $\Pr[\neg \mathcal{F}] \leq n^{-2}$ for all $t > n_1 = n^{\frac{11}{12}}$ according to Theorem 3.1. It is obvious that $D_t \geq i \cdot d_{t,s}(i)$, hence

$$\begin{aligned} E\left[\frac{d_{t,s}(i)}{D_t}\right] &= E\left[\frac{d_{t,s}(i)}{D_t} | \mathcal{F}\right] \Pr[\mathcal{F}] + E\left[\frac{d_{t,s}(i)}{D_t} | \neg \mathcal{F}\right] \Pr[\neg \mathcal{F}] \\ &\leq E\left[\frac{d_{t,s}(i)}{D_t} | \mathcal{F}\right] \Pr[\mathcal{F}] + \frac{\Pr[\neg \mathcal{F}]}{i}. \end{aligned}$$

CHAPTER 3. EVOLUTION OF SCALE-FREE NETWORKS WITH PARTIAL INFORMATION

Let $a = 2mS_t$ and $b = mS_t t^{\varphi-1}$. In the event $\mathcal{F}$, we have $a - b \leq D_t \leq a + b$. Thus, $\frac{1}{D_t} \leq \frac{1}{a-b}$ in this case. So

$$\begin{aligned} & \frac{1}{D_t} \left(\frac{a-b}{a} \right) \leq \left(\frac{1}{a-b} \right) \left(\frac{a-b}{a} \right) = \frac{1}{a} \\ \Rightarrow & \frac{1}{D_t} \left(1 - \frac{b}{a} \right) \leq \frac{1}{a} \\ \Rightarrow & \frac{1}{D_t} \leq \frac{1}{a} + \frac{b}{a} \frac{1}{D_t} = \frac{1}{2mS_t} + \frac{t^{\varphi-1}}{2} \frac{1}{D_t} \end{aligned}$$

Thus we have

$$\begin{aligned} E\left[\frac{d_{t,s}(i)}{D_t}\right] & \leq E\left[\frac{d_{t,s}(i)}{2mS_t} + \frac{d_{t,s}(i)}{D_t} \frac{t^{\varphi-1}}{2}\right] \Pr[\mathcal{F}] + \frac{\Pr[\neg\mathcal{F}]}{i} \\ & \leq \frac{E[d_{t,s}(i)]}{2mS_t} + \frac{t^{\varphi-1}}{2i} + \frac{\Pr[\neg\mathcal{F}]}{i}. \end{aligned}$$

Since $\Pr[\neg\mathcal{F}] \leq n^{-2}$ for all $t > n_1 = n^{\frac{11}{12}}$, we have

$$E\left[\frac{d_{t,s}(i)}{D_t}\right] \leq \frac{E[d_{t,s}(i)]}{2mS_t} + \frac{t^{\varphi-1}}{i}.$$

According to Lemma 3.1, $E[d_{t,s}(i)] = \frac{S_t}{t} E[d_t(i)]$. So if $t > n_1 = n^{\frac{11}{12}}$, we have whp

$$E\left[\frac{d_{t,s}(i)}{D_t}\right] = \frac{\bar{d}_t(i)}{2mt} + O\left(\frac{t^{\varphi-1}}{i}\right).$$

Similarly, we obtain

$$E\left[\frac{d_{t,s}(i-1)}{D_t}\right] = \frac{\bar{d}_t(i-1)}{2mt} + O\left(\frac{t^{\varphi-1}}{i-1}\right).$$

Thus, Eq. (3.1) can be converted into

$$\bar{d}_{t+1}(i) = \bar{d}_t(i) + \frac{\bar{d}_t(i-1)}{2t}(i-1) - \frac{\bar{d}_t(i)}{2t}i + O(t^{\varphi-1}),$$

for all $t > n_1 = n^{\frac{11}{12}}$.

This formula is similar to the recurrence equation in [36]. Thus, by a similar inductive analysis, we can get the following solution for all $1 \leq t \leq n$:

$$\bar{d}_t(i) = \frac{ct}{i(i+1)(i+2)} + O(n^\theta),$$

where c denotes a constant, and $\max\{\varphi, \frac{11}{12}\} < \theta < 1$.

■

3.4.1.3 Concentration of $d_t(i)$

Theorem 3.3 *Let m and β denote sufficiently large constants, and let $S_t = \beta \lg t$ for our model. Then there exists a constant $0 < \xi < 1$ such that*

$$\Pr [|d_n(i) - \bar{d}_n(i)| \geq n^\xi] \leq n^{-2}.$$

Proof: The proof can be obtained by applying a similar analysis of the concentration of D_t . ■

3.4.2 Maximum Degree

Theorem 3.4 *Let M_n denote the maximum degree of the graph G_n^m , then for any small constant $\epsilon \in (0, \frac{1}{2})$, whp*

$$C_2 n^{\frac{1}{2}-\epsilon} \leq M_n \leq C_1 n^{\frac{1}{2}+\epsilon},$$

where C_1 and C_2 denote constants.

Proof: It should be clear that by our construction, the degree of node 1 stochastically dominates the degrees of all the other nodes. Let X_t denote the degree of node 1 of the graph G_t^m at the time step t . Then we obtain

$$E[X_{t+1}|X_t] = X_t + \frac{mS_t}{t} \frac{X_t}{D_t}. \quad (3.2)$$

Initially, $X_0 = m$.

Let $x_t = E[X_t]$, then after taking the expectation on both sides of Eq.(3.2), we have

$$x_{t+1} = x_t + \frac{mS_t}{t} E\left[\frac{X_t}{D_t}\right] \quad (3.3)$$

Let $a = 2mS_t$, and $b = mS_t t^{\varphi-1}$, where φ denotes a constant and $0 < \varphi < 1$. We have the following relation, conditioned on the event that $|D_t - 2mS_t| < mS_t t^{\varphi-1}$.

$$\frac{1}{D_t} \leq \frac{1}{a-b} \leq \frac{1}{a} + \frac{b}{a^2} = \frac{1}{2mS_t} + \frac{mS_t t^{\varphi-1}}{2mS_t} = \frac{1}{2mS_t} + \frac{t^{\varphi-1}}{2}.$$

CHAPTER 3. EVOLUTION OF SCALE-FREE NETWORKS WITH PARTIAL INFORMATION

So Eq. (3.3) can be converted into

$$x_{t+1} \leq x_t + \frac{mS_t}{t} E\left[X_t \left(\frac{1}{2mS_t} + \frac{t^{\varphi-1}}{2}\right)\right] = \left(1 + \frac{1}{2t} + \frac{mS_t}{2t} t^{\varphi-1}\right) x_t.$$

Thus, for a constant $0 < \eta < 1$, we have

$$x_{t+1} \leq \left(1 + \frac{1}{2t} + t^{\eta-2}\right) x_t. \quad (3.4)$$

We now make the following hypothesis, and prove it inductively.

$$x_t \leq C_1 t^{\frac{1}{2}+\epsilon}, \quad (3.5)$$

where C_1 denotes a constant.

We argue that if Eq. (3.5) holds for $t \geq 1$, it also holds for $t + 1$. From Eq. (3.4), we have

$$x_{t+1} \leq C_1 \left(1 + \frac{1}{2t} + t^{\eta-2}\right) t^{\frac{1}{2}+\epsilon}.$$

Next, we are going to prove $x_{t+1} \leq C_1(t+1)^{\frac{1}{2}+\epsilon}$. Let $f(t) = C_1(t+1)^{\frac{1}{2}+\epsilon} - C_1\left(1 + \frac{1}{2t} + t^{\eta-2}\right) t^{\frac{1}{2}+\epsilon}$.

Since $(a+b)^n = a^n + \frac{n}{1!}a^{n-1}b + \frac{n(n-1)}{2!}a^{n-2}b^2 + \dots$,

$$\begin{aligned} f(t) = & C_1 t^{\frac{1}{2}+\epsilon} + C_1 \left(\frac{1}{2} + \epsilon\right) t^{\frac{1}{2}+\epsilon-1} + C_1 \frac{\left(\frac{1}{2} + \epsilon\right)\left(\frac{1}{2} + \epsilon - 1\right)}{2} t^{\frac{1}{2}+\epsilon-2} + h(t) \\ & - C_1 t^{\frac{1}{2}+\epsilon} - \frac{C_1}{2} t^{\frac{1}{2}+\epsilon-1} - C_1 t^{\frac{1}{2}+\epsilon+\eta-2}, \end{aligned}$$

where $h(t) = C_1(t+1)^{\frac{1}{2}+\epsilon} - C_1 t^{\frac{1}{2}+\epsilon} - C_1 \left(\frac{1}{2} + \epsilon\right) t^{\frac{1}{2}+\epsilon-1} - C_1 \frac{\left(\frac{1}{2} + \epsilon\right)\left(\frac{1}{2} + \epsilon - 1\right)}{2} t^{\frac{1}{2}+\epsilon-2} > 0$.

When $t \leq t_0 = \left(\frac{1}{\epsilon}\right)^{\frac{1}{1-\eta}}$, we can choose a sufficiently large C_1 to satisfy $f(t) \geq 0$. When $t > t_0 = \left(\frac{1}{\epsilon}\right)^{\frac{1}{1-\eta}}$, $\epsilon t^{\frac{1}{2}+\epsilon-1} - t^{\frac{1}{2}+\epsilon+\eta-2}$ is always positive. So $f(t) \geq 0$ for all $t \geq 1$. Thus

$$C_1(t+1)^{\frac{1}{2}+\epsilon} \geq \left(1 + \frac{1}{2t} + t^{\eta-2}\right) t^{\frac{1}{2}+\epsilon}.$$

So $x_{t+1} \leq C_1(t+1)^{\frac{1}{2}+\epsilon}$.

We can similarly show that $x_t \geq C_2 t^{\frac{1}{2}-\epsilon}$ for a constant C_2 .

Finally, we can argue that X_t concentrates around its expectation by applying a similar analysis of the concentration of D_t in Section 3.4.1.1. ■

3.4.3 Diameter

Theorem 3.5 *Let $\text{Diam}(G_n^m)$ denote the diameter of the graph G_n^m . Let $m \geq C_3 \lg n$ for a constant C_3 . Then **whp***

$$\text{Diam}(G_n^m) \leq 2 \lg n.$$

Proof: From Theorem 3.1, we know that **whp** $D_t \leq 2mS_t + mS_t t^{\varphi-1} \leq 3mS_t$. Since each node has degree at least m , the probability that a node in G_t^m is connected with the new node $t+1$ is at least

$$m \frac{S_t}{t} \frac{m}{D_t} \geq m \frac{S_t}{t} \frac{m}{3mS_t} = \frac{m}{3} \frac{1}{t}.$$

Consider the following graph $H_t^{\frac{m}{4}}$ based on the uniform attachment. At the first step, we start with a single node with $\frac{m}{4}$ self-loops. At the time step $t+1$, a new node $t+1$ is added into the graph $H_t^{\frac{m}{4}}$ to form $H_{t+1}^{\frac{m}{4}}$. It sends $\frac{m}{4}$ edges to the existing nodes chosen uniformly and randomly from $H_t^{\frac{m}{4}}$. As insightfully observed by Bollobás and Riordan in [15], we can argue that **whp** $H_t^{\frac{m}{4}}$ is a subgraph of G_t^m . Thus, $\text{Diam}(G_n^m) \leq \text{Diam}(H_n^{\frac{m}{4}})$. Next, we will show that **whp** $\text{Diam}(H_n^{\frac{m}{4}}) \leq 2 \lg n$.

We divide the nodes in $H_n^{\frac{m}{4}}$ into $\lg n$ sets, i.e., $B_i = \{x \mid 2^i \leq x < 2^{i+1}\}$, where $i = 0, \dots, \lg n$. Let $\mathcal{P}_i$ denote the event that each node in B_i is connected into nodes in B_{i-1} . We have

$$\Pr[\neg \mathcal{P}_i] \leq \left(\frac{1}{2}\right)^m 2^i.$$

Let $\mathcal{P} = \bigcap_{i=0}^{\lg n} \mathcal{P}_i$, then

$$\Pr[\neg \mathcal{P}] = \Pr\left[\bigcup_{i=0}^{\lg n} \neg \mathcal{P}_i\right] \leq \sum_{i=0}^{\lg n} \left(\frac{1}{2}\right)^m \cdot 2^i \leq \left(\frac{1}{2}\right)^m n \lg n.$$

When $m \geq C_3 \lg n$ for a sufficiently large constant C_3 , we can obtain $\Pr[\neg \mathcal{P}] < \frac{1}{n}$. So $\Pr[\text{Diam}(G_n^m) \geq 2 \lg n] < \frac{1}{n}$, as long as $m \geq C_3 \lg n$ for sufficiently large C_3 . ■

3.5 Experimental Evaluation

In this section, we experimentally evaluate the distribution of node degree, the maximum node degree and the network diameter of our proposed model. During the simulations, we vary the size of the sample set from 1 to n , where n denotes the current number of nodes in the network. At each run of the simulation, we also change the number of edges by each node and examine how this affects the final results.

Figure 3.1 plots the log-log curves of the number of nodes with different node degrees. Figure 3.1(a) shows the case when we only choose one node as a sample set, which is equal to the uniform attachment model [18]. Bollobás et al. [18] have mathematically shown that the proportion of nodes with different degree generally follows a geometric distribution. The case in Figure 3.1(f) with the total number of existing nodes as the sample set coincides with the BA model.

Our simulations show that when $|S_t| \geq 5$, the proportion P_k of nodes with degree k follows a power law distribution: $P_k \sim k^{-r}$, where $r \approx 3$ denotes as the slope of the log-log curve. Our experimental results indicate that our theoretical result (Result 1) is a little conservative. From the figures, we conjecture that when $|S_t| = \Omega(1)$, the distribution will obey a power law distribution.

Our experimental evaluations reveal that the change of m , the number of edges sent by each node, only affects the maximum node degree, but does not affect the value of r of the network, which means that it does not contribute to the emergence of the power law distribution of node degree.

From Figure 3.1, we can find that increasing the size of sample set does not affect the maximum degree in the network. Figure 3.2 plots the maximum node degree with different number of nodes, which verifies our second result, that is, the maximum degree is about $n^{0.5}$ for a network with n nodes.

3.6 Related Work

A few models based on only partial information have been proposed. Flaxman, Frieze, and Vera [36] presented a geometric preferential attachment model, in which every new node is generated at a random point in the unit sphere $\mathbf{R}^3$, and has only the connectivity information of its neighbors within certain distance. Each new node is connected to its reachable neighbors according to the PA rule. They reported that when the neighborhood and the number of edges established by each new node are sufficiently large, the degree of the whole graph will obey a power law. An empirical study of a one-dimensional geometric model with similar construction method was carried out by Gómez-Gardeñes and Moreno [43], which verified the power law in one dimension. The difference between our model and these models is that our partial information depends on the sampling information rather than the geometric neighborhood.

Two groups, Mossa et al. [78] and Štefančić et al. [100] have empirically studied the preferential attachment models with “information filtering”, which are quite similar to our model here. In their models of information filtering, each new node can process only a subset of randomly chosen existing nodes from the entire network. However, their results are entirely based on empirical studies and heuristic arguments, and without rigorous analysis, while our results are based on both rigorous analyses and experimental studies.

3.7 Concluding Remarks

In this chapter, we have proposed a new scale-free model for large-scale networks where a new joining node connects to some nodes in a small sample set; the connections follow the preferential attachment rule. We show that the power law of degree distribution still holds true. Compared with the existing models that require full network information, our construction based on partial information can better approximate the evolution of

CHAPTER 3. EVOLUTION OF SCALE-FREE NETWORKS WITH PARTIAL INFORMATION

large-scale distributed systems arising in the real world. Our results may also be applied to the constructions of peer-to-peer networks, where the global information is generally unavailable for any individual node in the network.

CHAPTER 3. EVOLUTION OF SCALE-FREE NETWORKS WITH PARTIAL INFORMATION

(a) $n = 10,000, |S_t| = 1.$

(b) $n = 10,000, |S_t| = 3.$

(c) $n = 10,000, |S_t| = 5.$

(d) $n = 10,000, |S_t| = 7.$

(e) $n = 10,000, |S_t| = 5000.$

50

(f) $n = 10,000, |S_t| = n.$

Figure 3.1: The distribution of node degree with different sizes of sample set.

CHAPTER 3. EVOLUTION OF SCALE-FREE NETWORKS WITH PARTIAL INFORMATION

Figure 3.2: Maximum node degree in networks with different numbers of nodes.

Chapter 4

Near-Optimal Routing for Small-World Networks with Augmented Local Awareness

In order to investigate the routing aspects of small-world networks, Kleinberg [61] proposes a network model based on a d -dimensional lattice with long-range links chosen at random according to the d -harmonic distribution. Kleinberg shows that the greedy routing algorithm by using only local information performs in $O(\lg^2 n)$ expected number of hops, where n denotes the number of nodes in the network. Martel and Nguyen [72] have found that the expected diameter of Kleinberg's small-world networks is $\Theta(\lg n)$. Thus a question arises naturally: Can we improve the routing algorithms to match the diameter of the networks while keeping the amount of information stored on each node as small as possible?

Existing approaches for improving the routing performance in the small-world networks include: (1) Increasing the number of long-range links [8]; (2) Exploring more nodes before making routing decisions [65]; (3) Increasing the local awareness for each node [39, 72]. However, all these approaches can only achieve $O((\lg n)^{1+\epsilon})$ expected number of hops, where $\epsilon > 0$ denotes a constant. We will extend Kleinberg's model and add three augmented local links for each node: two of which are connected to nodes chosen randomly and uniformly within $\lg^2 n$ Manhattan distance, and the third one is connected to a node chosen

randomly and uniformly within $\lg n$ Manhattan distance. Our investigation will show that these augmented local connections can make small-world networks more navigable.

We will show that if each node is aware of $O(\lg n)$ number of neighbors via the augmented local links, there exist both non-oblivious and oblivious algorithms that can route messages between any pair of nodes in $O(\lg n \lg \lg n)$ expected number of hops, which outperforms the other related results for Kleinberg's small-world networks. Our schemes keep only $O(\lg^2 n)$ bits of routing information on each node. Thus our schemes are scalable with the network size.

The chapter is organized as follows. Section 4.1 gives a brief introduction. Section 4.2 gives notations for Kleinberg's small-world model and its extended version with augmented local connections. Section 4.3 gives some preliminary notations for decentralized routing. In Section 4.3, we propose both non-oblivious and oblivious schemes with near-optimal routing complexity in our extended model. Section 4.5 briefly concludes this chapter.

4.1 Introduction

Here we briefly recall the related work in this area. A well-known study by Milgram in 1967 [93] shows the *small-world phenomenon* [32], also called “six degree of separation”, that any two people in the world can be connected by a chain of six (on the average) acquaintances, and people can deliver messages efficiently to an unknown target via their acquaintances. This study is repeated by Dodds, Muhamad, and Watts [28] recently, and the results show that it is still true for today's social network. The small-world phenomenon has also been shown to be pervasive in networks from nature and engineering systems, such as the World Wide Web [27, 4], peer-to-peer systems [8, 71, 70, 106], etc.

Recently, a number of network models have been proposed to study the small-world properties [74, 27, 61]. Watts and Strogatz [27] propose a random rewiring model whose diameter is a poly-logarithmic function of the size of the network. The model is constructed

CHAPTER 4. NEAR-OPTIMAL ROUTING FOR SMALL-WORLD NETWORKS WITH AUGMENTED LOCAL AWARENESS

by adding a small number of random edges to nodes uniformly distributed on a ring, where nodes are connected densely with their near neighbors. A similar approach can also be found in Ballabás and Chung’s earlier work [9], where the poly-logarithmic diameter of the random graph is achieved by adding a random matching to the nodes of a cycle. However, these models fail to capture the algorithmic aspects of a small-world network [61]. As commented by Kleinberg in [61], the poly-logarithmic diameter of some graphs does not imply the existence of efficient routing algorithms. For example, the random graph in [9] yields a logarithmic diameter, yet any routing using only local information requires at least $\sqrt{n}$ expected number of hops (where n is the size of the network) [61]. In order to address the algorithmic aspects, Kleinberg develops a new model based on a d -dimensional torus lattice with long-range links chosen randomly from the d -harmonic distribution. Based on this model, Kleinberg then shows that routing messages between any two nodes can be achieved in $O(\lg^2 n)$ expected number of hops by applying a simple greedy routing algorithm using only local information.

Kleinberg’s model has generated considerable interests, e.g. [11, 65, 72, 39, 71, 48]. The reader is referred to Chapter 2 for more details.

4.1.1 Our Contributions

We extend Kleinberg’s structures of small-world models with slight change. Beside having long-range and local links on the grid lattice, each node is augmented with three extra links, two of which are chosen randomly and uniformly from nodes within $\lg^2 n$ Manhattan distance, and the third one is chosen randomly and uniformly from nodes within $\lg n$ Manhattan distance. Based on this extended model, we present near-optimal algorithms for decentralized routing with $O(\lg n)$ augmented awareness. We show that if each node is aware of $O(\lg n)$ number of augmented neighbors via lookahead, there exist both non-oblivious and oblivious routing algorithms that perform in $O(\lg n \lg \lg n)$ expected number

CHAPTER 4. NEAR-OPTIMAL ROUTING FOR SMALL-WORLD NETWORKS WITH AUGMENTED LOCAL AWARENESS

Scheme	#bits of awareness	#steps expected	Oblivious or Non-oblivious?
Kleinberg's greedy [61, 8, 11]	$O(q \lg n)$	$O(\lg^2 n / q)$	Oblivious
NoN-greedy [71]	$O(q^2 \lg n)$	$O(\lg^2 n / (q \lg q))$	Non-oblivious
Decentralized algorithm in [65]	$O(\lg^2 n / \lg(1 + q))$	$O((\lg n)^2 / \lg^2(1 + q))$	Non-oblivious
Decentralized algorithm [72]	$O(\lg^2 n)$	$O((\lg n)^{1+1/d})$	Non-oblivious
Indirect-greedy algorithm [39]	$O(\lg^2 n)$	$O((\lg n)^{1+1/d})$	Oblivious
Our algorithms for the model with augmented awareness	$O(\lg^2 n)$	$O(\lg n \lg \lg n)$	Both are provided

Table 4.1: Comparisons of our decentralized routing algorithms with other existing schemes.

of hops (see Theorem 4.1 and Theorem 4.2). Our investigation constructively shows that the augmented local connections can make small-world networks more navigable.

A comparison of our algorithm with the other existing schemes is shown in Table 5.1. In the first three schemes (in [61, 8, 11, 71, 65]), we suppose that each node has q long-range contacts, while in the next three schemes (in [72, 39] and this chapter), we suppose that each node has one long-range contact. Our decentralized routing algorithms assume that each node can compute a shortest path among a poly-logarithmic number of known nodes. Such an assumption is reasonable since each node in a computer network is normally a processor and can carry out such a simple computation. Our schemes keep a poly-logarithmic number of bits of routing information stored on each node, thus they are scalable as the network size increases.

We note that beside shedding new light to the studies of social networks such as Milgram's experiment [93], our results may also find applications in the design of large-scale distributed networks, such as peer-to-peer systems, in the same spirit of Symphony [70]. Since the links in our extended model are randomly constructed according to the probabilistic distribution, the network is less vulnerable to adversarial attacks, and thus provides good fault tolerance.

4.2 Definitions of Small-World Models

In this section, we will give definition of Kleinberg's small-world model and its extended version in which each node has extra links. For simplicity, we only consider the one-

CHAPTER 4. NEAR-OPTIMAL ROUTING FOR SMALL-WORLD NETWORKS WITH AUGMENTED LOCAL AWARENESS

dimensional model with *one* long-range contact for each node. The results for multiple-dimensional models could be easily obtained. In addition, we assume that all links are directed, which is consistent with the real-world observation, for example, person x knows person y , but y may not know x .

Definition 4.1 (Kleinberg’s Small-World Network (KSWN) [61]) *A Kleinberg’s Small-World Network, denoted as $\mathcal{K}$, is based on a one-dimensional torus (or ring) $[n] = [0, 1, \dots, n]$. Each node u has a directed local link to its next neighbor $(u + 1) \bmod n$ on the ring. We refer to this local link as **Ring-link** (or **R-link** for short), and refer to node $(u + 1) \bmod n$ as the **R-neighbor** of node u . In addition, each node has one long-range link to another node chosen randomly according to the 1-harmonic distribution, that is, the probability that node u sends a long-range link to node v is $\Pr[u \rightarrow v] = \frac{1}{Z_u \cdot \text{Dist}(u, v)}$, where $\text{Dist}(u, v)$ denotes the ring distance¹ from u to v , and $Z_u = \sum_{z \neq u} \frac{1}{\text{Dist}(u, z)}$. We refer to this long-range link as the **Kleinberg-link** (or **K-link** for short), and refer to node v as a **K-neighbor** of node u if a K-link exists from u to v .*

Our extended structure introduces several extra links for each node. Its definition is given below.

Definition 4.2 (KSWN with Augmented Local Awareness (KSWN*)) *A Kleinberg’s Small-World Network with Augmented Local Awareness, denoted as $\mathcal{K}^*$, has the same structure of KSWN, except that each node u in $\mathcal{K}^*$ has three extra links. First, node u has two links to nodes chosen randomly and uniformly from the interval $(u, u + \lg^2 n]$. We refer to these two links as the **primary augmented local links** (or **PAL-links** for short), and refer to node v as a **PAL-neighbor** of node u if a PAL-link exists from u to v . In addition, each node u has one link to a node chosen randomly and uniformly from the interval $(u, u + \lg n]$. We refer to this link as the **secondary augmented local link** (or*

¹or Manhattan distance for multi-dimensional models.

SAL-link for short), and refer to node v as a **SAL-neighbor** of node u if a SAL-link exists from u to v .

There are in total five links for each node in a KSWN*: one R-link, one K-link, two PAL-links and one SAL-link. We refer to the nodes linked directly by node u as the **immediate neighbors** of u . Our extended structure retains the same $O(1)$ order of node degree as that of Kleinberg's original model.

4.3 Decentralized Routing Algorithms

Based on the original model, Kleinberg presents a class of decentralized routing algorithms, in which each node makes routing decisions by using local information and in a greedy fashion. In other words, the message holder forwards the message to its immediate neighboring node, including its K-neighbor, which is closest to the destination in terms of the Manhattan distance. Kleinberg shows that such a simple greedy algorithm performs in $O(\lg^2 n)$ expected number of hops. The other existing decentralized routing algorithms [8, 65, 39, 72, 71] rely on three approaches to improve routing performance: (1) Increasing the number of long-range links [8]; (2) Exploring more nodes before making routing decisions [65]; (3) Increasing the local awareness for each node [39, 72]. However, so far using these approaches can only achieve $O((\lg n)^{1+\epsilon})$ expected number of steps in routing, where $\epsilon = 1$ or $\epsilon = \frac{1}{d}$. Although the scheme in [71], where each node makes routing decision by looking ahead its neighbors's neighbors, can achieve an optimal $O(\lg n / \lg \lg n)$ bound, their result depends on the fact that each node has at least $\Omega(\lg n)$ number of K-links.

There are normally two approaches for decentralized routing: oblivious and non-oblivious schemes [39]. A routing protocol is *oblivious* if the message holder makes routing decisions only by its local information and the target node, and independently of the previous routing history. On the other hand, if the message holder need to consider certain information

CHAPTER 4. NEAR-OPTIMAL ROUTING FOR SMALL-WORLD NETWORKS WITH AUGMENTED LOCAL AWARENESS

of the previous routing history to make routing decisions, the protocol is referred to as *non-oblivious*. The non-oblivious protocol is often implemented by adding a header segment to the message packet so that the downstream nodes can learn the routing decisions of upstream nodes by reading the message header information. The scheme in [39] is oblivious, while the schemes in [65] and [72] are non-oblivious.

For a given node x , we define a sequence of node sets $T_0, T_1, \dots, T_i, \dots$, where $T_0 = \{x\}$, $T_1 = \{u \text{'s PAL-neighbors}, \forall u \in T_0\}$, $T_2 = \{u \text{'s PAL-neighbors}, \forall u \in T_1\}$, and so on. We refer to T_i as the set of nodes during *the i th level of PAL lookahead*, and let $H_i = \bigcup_{j \leq i} T_j$ denote the set of all nodes in *the first i levels of PAL lookahead*. At a certain level i of lookahead, we may also refer to H_{i-1} as the set of *previously known nodes*. Let $L_i = T_i - H_{i-1}$ denote the set of *new* nodes discovered by the i th level of PAL lookahead. Let $A_x(k) = H_k$ denote the *primary augmented local awareness* (or *PAL awareness* for short) of a given KSWN*, where each node can look ahead k levels of its PAL-neighbors.

In Section 4.4, we will show that $|A_x(\lg \lg n)| \geq \lg n / \sigma$ for a constant σ , based on which we propose both non-oblivious and oblivious routing algorithms running in $O(\lg n \lg \lg n)$ expected number of hops and requiring $O(\lg^2 n)$ bits of information on each node.

Our near-optimal $O(\lg n \lg \lg n)$ bound on the routing complexity outperforms the other related results for Kleinberg's small-world networks. Also, the number of $\lg \lg n$ levels of lookahead in our schemes is small compared with that in [39, 72], which requires $(\lg n)^{1/d}$ levels of lookahead in local links for a d -dimensional model.

4.4 near-optimal Routing with $O(\lg n)$ Awareness

4.4.1 Primary Augmented Local Awareness of $O(\lg n)$

In this subsection, we will show that $|A_x(\lg \lg n)|$, the number of distinct nodes that node x is aware of through $\lg \lg n$ levels of PAL lookahead, is not less than $\lg n / \sigma$ for a constant σ , which, as will be shown in Lemma 4.3, is sufficiently large to guarantee that $A_x(\lg \lg n)$

CHAPTER 4. NEAR-OPTIMAL ROUTING FOR SMALL-WORLD NETWORKS WITH AUGMENTED LOCAL AWARENESS

contains a K-link that jumps over half distance (Suppose that the destination node is at a certain large distance from the current node). These results are useful for the subsequent analysis of our oblivious and non-oblivious routing schemes.

Lemma 4.1 *Let $A_x(\lg \lg n)$ denote the PAL awareness of node x in a $\text{KSWN}^* \mathcal{K}^*$, where each node is aware of $\lg \lg n$ levels of PAL-neighbors. Then*

$$\Pr[|A_x(\lg \lg n)| \geq \frac{\lg n}{\sigma}] > \psi,$$

where σ denotes a sufficiently large constant and ψ denotes a positive constant.

Proof: Throughout the proof, we assume that $|H_i| < \frac{\lg n}{\sigma}$ for all $1 \leq i \leq \lg \lg n$, otherwise, the lemma already holds, since $|A_x(\lg \lg n)| = |H_{\lg \lg n}| > \lg n / \sigma$. We will show that at each level of lookahead, the probability that each PAL-link points to previously known nodes is small so that a large number of distinct nodes will be found by $\lg \lg n$ levels of PAL lookahead.

Consider the construction of a PAL-link for a given node x . By definition of KSWN^* , each PAL-link of x is connected to a node randomly and uniformly chosen from the interval $(x, x + \lg^2 n]$, that is, each PAL-link of x points to a node in the interval $(x, x + \lg^2 n]$ with probability $(\lg n)^{-2}$. By assumption, there could be no more than $\lg n / \sigma$ previously known nodes in the interval $(x, x + \lg^2 n]$. Thus, the probability for a PAL-link of a given node to point to a previously known node is at most $(\lg n / \sigma) \cdot (\lg n)^{-2} = (\sigma \lg n)^{-1}$. Thus, the probability for a PAL-link of a given node to point to a new node is at least $1 - (\sigma \lg n)^{-1}$. There are in total at most $2 \cdot |H_{\lg \lg n}| \leq 2 \lg n / \sigma$ number of PAL-links, so the probability for all PAL-links to point to new nodes is at least $(1 - (\sigma \lg n)^{-1})^{2 \lg n / \sigma} \geq 1 - \frac{2}{\sigma^2}$ for sufficiently large n . Here we use the fact $(1 + x)^a \geq 1 + ax$ for $x > -1$ and $a \geq 1$. When σ is a sufficiently large constant, we have $\Pr[|A_x| \geq \frac{\lg n}{\sigma}] > \psi$ for a positive constant $\psi = 1 - \frac{2}{\sigma^2} > 0$. Thus, the proof of Lemma 4.1 is completed. ■

4.4.2 Non-Oblivious Decentralized Routing

Our non-oblivious routing algorithm is given as follows: Initially the source node s finds in its PAL awareness $A_s(\lg \lg n)$ an intermediate node z that is closest to the destination, and then computes a shortest path π from s to z in $A_s(\lg \lg n)$. Before routing the message, s adds the information about shortest path π to the message header. Once the message passes a node on the shortest path π , the next stop is read off the header stack. When the message reaches node z , node z can tell that it is an intermediate target by reading the message header and then route the message to its K -neighbor. Such processes are repeated until the message reaches a certain distance to the destination node. After that, Kleinberg's plain greedy algorithm can be used to route the message effectively to the target node. Given a message M , a source node s and a target node t in a KSWN* $\mathcal{K}^*$, the pseudocodes of our non-oblivious algorithm running on node x are given as follows.

Algorithm 1

Input: the source s , the target t and the message M .

Initialization:

```

 $x \leftarrow s$ .
Set the header stack of the message  $M$  to be empty.
while  $\text{Dist}(x, t) \geq (\lg n)^2 \lg \lg n$  do
  if the header stack of the message  $M$  is empty then
    Route the message  $M$  to  $x$ 's  $K$ -neighbor  $y$ .
    Find an intermediate node  $z$  in  $A_y(\lg \lg n)$  whose  $K$ -neighbor is closest to  $t$  (ties are broken arbitrarily).
    Compute a shortest path  $\pi : x_0 = y, x_1, \dots, x_t = z$  from  $x$  to  $z$ , and push the shortest path information  $\pi : x_1, \dots, x_t = z$  into the header stack of the message  $M$ .
  else
    Pop up the first node  $x_i$  from the header stack and route the message  $M$  to node  $x_i$ .
  end if
end while

```

Final phase (Kleinberg's greedy algorithm):

Route the message M to an immediate neighbor of x that is closest to the target t , until it reaches t .

Next we will analyze the performance of the Algorithm 1. We first give a basic lemma, which provides a lower bound and an upper bound on the probability of the existence of a K -link in Kleinberg's small-world networks.

Lemma 4.2 *Let $\Pr[u \xrightarrow{K} v]$ denote the probability that node u sends a K -link to node v in a KSWN* $\mathcal{K}^*$. Suppose that $a \leq \text{Dist}(u, v) \leq b$, then $\frac{c_1}{b \lg n} \leq \Pr[u \xrightarrow{K} v] \leq \frac{c_2}{a \lg n}$, where c_1 and c_2 are constants independent of n .*

CHAPTER 4. NEAR-OPTIMAL ROUTING FOR SMALL-WORLD NETWORKS WITH AUGMENTED LOCAL AWARENESS

Proof: The probability that node v is a K -neighbor of node u is $\Pr[u \xrightarrow{K} v] = \frac{1}{\text{Dist}(u,v)Z_v}$, where $\text{Dist}(u,v)$ is the ring distance between nodes u and v , and $Z_v = \sum_{z \neq v} \frac{1}{\text{Dist}(v,z)}$.

Observe that $Z_v = \sum_{i=1}^n \frac{|U_i|}{i}$, where $|U_i|$ is the set of all nodes at distance i away to node v . Since $|U_i| = \Theta(1)$, we have $Z_v = \sum_{i=1}^n \frac{\Theta(1)}{i} = \Theta(\lg n)$.

Since $a \leq \text{Dist}(u,v) \leq b$, we have $\frac{c_1}{b \lg n} < \Pr[u \xrightarrow{K} v] < \frac{c_2}{a \lg n}$, for some constants c_1 and c_2 independent of n . Thus the lemma follows. $\blacksquare$

In Lemma 4.1, we have shown that $\Pr[|A_x(\lg \lg n)| \geq \lg n / \sigma]$ is at least a positive constant for a sufficiently large constant σ . Based on this result, Lemma 4.3 shows that the probability for $A_x(\lg \lg n)$ to contain a K -link jumping over $\text{Dist}(x,t)/2$ is at least a positive constant.

Lemma 4.3 *Suppose that the distance between the current node x and the target node t in a $KSWN^*K^*$ is $\text{Dist}(x,t) \geq \lg^2 n \lg \lg n$. Then with probability at least a positive constant, node x 's PAL awareness $A_x(\lg \lg n)$ contains a K -neighbor within $\text{Dist}(x,t)/2$ distance to the target node t .*

Proof: Let $\mathcal{A}$ denote the event that $|A_x(\lg \lg n)| \geq \frac{\lg n}{\sigma}$. By Lemma 4.1, we have $\Pr[\mathcal{A}] > \psi$ for a constant $\psi > 0$.

Let $B_l(t)$ denote the set of all nodes within l ring distance to t . Let $\Pr[x \xrightarrow{K} B_l(t)]$ denote the probability that x 's K -neighbor is inside the ball $B_l(t)$.

Let $m = \text{Dist}(x,t)$. By Lemma 4.2, the probability for a K -link to point to a given node inside the ball $B_{\frac{m}{2}}(t)$ is at least $\frac{c_1}{m \lg n}$, so we have

$$\Pr[x \xrightarrow{K} B_{\frac{m}{2}}(t)] \geq |B_{\frac{m}{2}}(t)| \cdot \frac{c_1}{m \lg n} = \frac{m}{2} \cdot \frac{c_1}{m \lg n} \geq \frac{c_3}{\lg n},$$

where c_3 is a constant.

Since $\text{Dist}(x,t) \geq \lg^2 n \lg \lg n$ and each PAL-link spans a distance no more than $\lg^2 n$, the nodes in PAL awareness $A_x(\lg \lg n)$ are all between the current node x and the target

CHAPTER 4. NEAR-OPTIMAL ROUTING FOR SMALL-WORLD NETWORKS WITH AUGMENTED LOCAL AWARENESS

node t . Let $\Pr[A_x \xrightarrow{K} B_{\frac{m}{2}}(t)]$ denote the probability that at least one node in $A_x(\lg \lg n)$ has a K -neighbor in $B_{\frac{m}{2}}(t)$. Then we have

$$\begin{aligned} \Pr[A_x(\lg \lg n) \xrightarrow{K} B_{\frac{m}{2}}(t)] &\geq \Pr[A_x(\lg \lg n) \xrightarrow{K} B_{\frac{m}{2}}(t) \mid \mathcal{A}] \cdot \Pr[\mathcal{A}] \\ &\geq \left(1 - \left(1 - \frac{c_3}{\lg n}\right)^{\frac{\lg n}{\sigma}}\right) \cdot \psi \\ &\geq \psi(1 - e^{-\frac{c_3}{\sigma}}), \end{aligned}$$

which is larger than a positive constant. At the last step, we obtain $(1 - \frac{c_3}{\lg n})^{\frac{\lg n}{\sigma}} \leq e^{-\frac{c_3}{\sigma}}$ by using the fact that $(1 + \frac{b}{x})^x \leq e^b$ for $b \in \mathbb{R}$ and $x > 0$. ■

Lemma 4.4 *Suppose that the distance between the current node x and the target node t in a $KSWN^* \mathcal{K}^*$ is $\text{Dist}(x, t) \geq \lg^2 n \lg \lg n$. Then after at most $O(\lg n \lg \lg n)$ expected number of hops, Algorithm 1 will reduce the distance to within $\lg^2 n \lg \lg n$.*

Proof: Since $\text{Dist}(x, t) \geq \lg^2 n \lg \lg n$, all known nodes in x 's PAL awareness $A_x(\lg \lg n)$ are between the current node x and the target node t . We can apply the result in Lemma 4.3 to analyze Algorithm 1.

We refer to the routing steps from a given node x to its intermediate node z in $A_x(\lg \lg n)$ as an indirect phase. The routings in different indirect phases are independent from each other. By Lemma 4.3, the probability that node x 's PAL awareness $A_x(\lg \lg n)$ has a K -neighbor within $\frac{\text{Dist}(x, t)}{2}$ distance to the target node t is at least a positive constant, so after at most $O(1)$ expected number of indirect phases, Algorithm 1 will find an intermediate node whose K -link jump over half distance. Since each indirect phase takes at most $\lg \lg n$ hops and the maximum distance is n , after at most $O(\lg n \lg \lg n)$ expected number of hops, the message will reach a node within $\lg^2 n \lg \lg n$ distance to the target node t . ■

Lemma 4.5 *Suppose that the distance between the current node x and the target node t in a $*KSWN^* \mathcal{K}$ is $\text{Dist}(x, t) \leq \lg^2 n \lg \lg n$. Then after at most $O(\lg n)$ expected number of hops, the message will reach the target node t .*

Proof: When the distance $Dist(x, t) \leq \lg^2 n \lg \lg n$, the final phase in Algorithm 1 is executed.

First, since each PAL-neighbor is chosen uniformly and randomly from the interval $(x, x + \lg^2 n]$, the probability for a PAL-link to jump over $\frac{\lg^2 n}{2}$ distance is $\frac{1}{2}$, that is, after $O(1)$ number of hops, the routing distance will be reduced by $\frac{\lg^2 n}{2}$. Thus, after at most $O(\lg \lg n)$ expected number of hops, the routing distance will be reduced to within $\lg^2 n$. Next, by applying a similar method, we can also show that after $O(\lg n)$ expected number of hops via secondary augmented local links (SAL-links), the routing distance will be reduced to within $\lg n$. Finally, using the simple greedy algorithm via the R-links can route the message to the target node in $\lg n$ hops. Therefore, the greedy routing algorithm only by using the information of immediate neighbors can finish the routing in $O(\lg n)$ expected number of hops. ■

Combining the above lemmas, it is not difficult for us to obtain the routing complexity of Algorithm 1.

Theorem 4.1 *In a KSWN* $\mathcal{K}^*$, Algorithm 1 performs in $O(\lg n \lg \lg n)$ expected number of hops.*

4.4.3 Oblivious Decentralized Routing

In our oblivious scheme, when the distance is large, the current node x first finds in $A_x(\lg \lg n)$ whether there is an intermediate node z , which contains a K-neighbor within $\frac{Dist(t, x)}{2}$ distance to the target node, and is closest to node x in terms of PAL-links (any possible tie is broken arbitrarily). Next, node x computes a shortest path π from x to z , and then routes the message to its next PAL-neighbor on the shortest path π . When the distance is small, Kleinberg's plain greedy is applied.

Given a message M , a source s and a target t in a KSWN* $\mathcal{K}^*$, the pseudocodes of our oblivious algorithm running on node x are given in Algorithm 2.

CHAPTER 4. NEAR-OPTIMAL ROUTING FOR SMALL-WORLD NETWORKS WITH AUGMENTED LOCAL AWARENESS

Algorithm 2

Input: the source s , the target t and the message M .

Initialization:

```

 $x \leftarrow s$ .
while  $\text{Dist}(x, t) \geq c(\lg n)^2 \lg \lg n$  ( $c$  is a sufficiently large constant and will be given later) do
   $z \leftarrow$  a node in  $A_x(\lg \lg n)$  that contains a K-neighbor within  $\frac{\text{Dist}(t, x)}{2}$  distance to  $t$ , and is closest to node  $x$  in terms of
  PAL-links (ties are broken arbitrarily).
  if node  $z$  does not exist then
    Route the message  $M$  to an immediate neighbor closest to node  $t$ .
  else
    Compute a shortest path  $\pi$  from  $x$  to  $z$  among  $A_x(\lg \lg n)$ .
    if  $\pi$  consists of only node  $x$  itself then
      Route the message  $M$  to the K-neighbor.
    else
      Route the message  $M$  to the next PAL-neighbor on the shortest path  $\pi$ .
    end if
  end if
end while

```

Final phase (Kleinberg's greedy algorithm):

Route the message M to an immediate neighbor of x that is closest to the target t , until it reaches t .

Lemma 4.6 *Suppose that the distance between the current node x and the target node t in a $KS\text{WN}^* K^*$ is $\text{Dist}(x, t) \geq c(\lg n)^2 \lg \lg n$, where c is a sufficiently large constant. Then after at most $O(\lg \lg n)$ expected number of hops, Algorithm 2 will reduce the distance to $\text{Dist}(x, t)/2$.*

Proof: As shown in Figure 4.1, node r is the midpoint of $\overline{xt}$, and node r' is between r and t such that $\text{Dist}(r, r') = \lg^2 n \lg \lg n$. Let z be an intermediate node in $A_x(\lg \lg n)$ that contains a K-neighbor between r and t , and is closest to x in terms of AL-links. We refer to a node z in x 's AL awareness $A_x(\lg \lg n)$ as a *good* intermediate node if it satisfies the following two conditions: (1) has a K-neighbor within $\text{Dist}(x, t)/2$ to the target node; (2) is closest to node x in terms of AL-links. Let $\pi : x_0 = x, x_1, \dots, x_t = z$ denote a shortest path that x finds from itself to z among the AL awareness $A_x(\lg \lg n)$. We divide the next routing into two cases according to the different locations of z 's K-neighbor.

In the first case, z 's K-neighbor is within $\overline{r't}$. Since the distance between x and the right most node in $A_x(\lg \lg n)$ is at most $\lg^2 n \lg \lg n$, z 's K-neighbor is also within $\text{Dist}(x_i, t)/2$ to the target node for every x_i on the shortest path π , that is, node z always satisfies the first condition of a good intermediate node for every node x_i . Also, if z is an intermediate node closest to x , it is also a closest intermediate node to every x_i on the shortest path

CHAPTER 4. NEAR-OPTIMAL ROUTING FOR SMALL-WORLD NETWORKS WITH AUGMENTED LOCAL AWARENESS

π , that is, z also satisfies the second condition of a good intermediate node for every node x_i . Therefore, node z will become a fixed good intermediate node for all nodes x_i on the shortest path. When this case happens, Algorithm 2 will route the message along a shortest path from x to z in an oblivious routing fashion. Thus, in this case, after at most $\lg \lg n$ number of hops, the message will reach a good intermediate node and the routing distance will be reduced by half². In the second case, z 's K-neighbor is within $\overline{rr'}$. When this happens, the intermediate node z may change for each x_i on the shortest path $\pi : x_0 = x, x_1, \dots, x_t = z$, and the message may not be routed along the shortest path as expected by the previous node x . However, we will show that the latter case will not happen very likely, since the length of $\overline{rr'}$ is relatively small when $\text{Dist}(x, t) \geq c(\lg n)^2 \lg \lg n$ for a sufficiently large constant c .

Let $\mathcal{F}_1$ denote the event that $A_x(\lg \lg n)$ contains a K-neighbor in $\overline{rr'}$. By using a similar technique in Lemma 4.3, we can easily obtain that $\mathcal{F}_1$ occurs with probability at least a positive constant.

Let $\mathcal{F}_2$ denote the event that $A_x(\lg \lg n)$ contains a K-neighbor in $\overline{rr'}$. For any node y in $A_x(\lg \lg n)$, we have $\text{Dist}(y, r) \geq \frac{1}{3}c(\lg n)^2 \lg \lg n$ when c is a sufficiently large constant. By Lemma 4.2, the probability for a node y in $A_x(\lg \lg n)$ to send a K-link to a node in $\overline{rr'}$ is at most $\frac{3c_2}{c(\lg n)^2(\lg \lg n) \cdot \lg n}$. Because there are in total $\lg^2 n \lg \lg n$ nodes in $\overline{rr'}$, a node in $A_x(\lg \lg n)$ has a K-neighbor in $\overline{rr'}$ with probability at most $\frac{3c_2}{c(\lg n)^2(\lg \lg n) \cdot \lg n} \cdot \lg^2 n \lg \lg n = \frac{3c_2}{c \lg n}$. Since $|A_x(\lg \lg n)| \leq 1 + 2 + 2^2 + \dots + 2^{\lg \lg n} \leq 2 \lg n$, the event $\mathcal{F}_2$, i.e., $A_x(\lg \lg n)$ has a K-neighbor in $\overline{rr'}$, occurs with probability at most $\frac{3c_2}{c \lg n} \cdot 2 \lg n = \frac{6c_2}{c}$, which is smaller than a certain constant when c is a sufficiently large constant. Thus, we have $\Pr[\neg \mathcal{F}_2] > \gamma$ for a constant $\gamma > 0$, if we choose a sufficiently large constant c .

Therefore, $\Pr[\neg \mathcal{F}_2 \cap \mathcal{F}_1]$ is larger than a positive constant, if we choose a sufficiently large constant c . Thus, after at most $c' \lg \lg n$ expected number of hops for a constant c' ,

²There may be more than one good intermediate nodes z when tie happens. However, even this happens, the message will still be routed to one of good intermediate nodes along a shortest path finally.

CHAPTER 4. NEAR-OPTIMAL ROUTING FOR SMALL-WORLD NETWORKS WITH AUGMENTED LOCAL AWARENESS

Figure 4.1: Diagram for oblivious decentralized routing.

the event $\neg \mathcal{F}_2 \cap \mathcal{F}_1$ will occur, that is, a message will be routed to a node x whose PAL awareness $A_x(\lg \lg n)$ contains a K-neighbor in $\overline{r't}$, but no K-neighbor in $\overline{rr'}$. When such a node x is reached, the intermediate node z is fixed for every node x_i on a shortest path $\pi : x_0 = x, x_1, \dots, x_t = z$ in an obvious routing fashion. Then after at most $\lg \lg n$ number of hops, the message will be routed to the fixed intermediate node z , which has a K-link jumping over half distance.

Therefore, after at most $c' \lg \lg n + \lg \lg n = O(\lg \lg n)$ expected number of hops, the distance will be reduced by half. ■

Lemma 4.7 Suppose that the distance between the current node x and the target node t in a $KSWN^* K^*$ is $\text{Dist}(x, t) \geq c \lg^2 n \lg \lg n$, where c is a sufficiently large constant. Then after at most $O(\lg n \lg \lg n)$ expected number of hops, Algorithm 2 will reduce the routing distance to within $c \lg^2 n \lg \lg n$.

Proof: The proof is similar to that of Lemma 4.4, and hence is omitted here. ■

Lemma 4.8 Suppose that the distance between current node x and the target node t in a $KSWN^* K^*$ is $m < c(\lg n)^2 \lg \lg n$, where c is a sufficiently large constant. Then after at most $O(\lg n)$ expected number of hops, Algorithm 2 will route the message to the target node t .

Proof: The proof is similar to that of Lemma 4.5, and hence is omitted here. ■

Combining the above lemmas, we can easily obtain the following theorem.

Theorem 4.2 *In a $KSWN^*K^*$, Algorithm 2 performs in $O(\lg n \lg \lg n)$ expected number of hops.*

4.5 Concluding Remarks

We extend Kleinberg’s small-world network with augmented local links, and show that if each node participating in the routing is aware of $O(\lg n)$ neighbors via augmented links, there exist both non-oblivious and oblivious decentralized algorithms that can finish routing in $O(\lg n \lg \lg n)$ expected number of hops, which is a near-optimal routing complexity. Our investigation shows that the awareness of $O(\lg n)$ nodes through the augmented links will be more efficient for routing rather than via the local links [39, 72].

Our extended model may provide an important supplement for the modelling of small-world phenomenon, and may better approximate the real-world observation. For example, each person in a human society is very likely to increase his/her activities randomly within some certain communities, and thus is aware of certain levels of “augmented” acquaintances. This augmented awareness would surely help delivery the message between any two persons in the society.

Our results may also find applications in the design of large-scale distributed networks, such as distributed storage systems. Unlike most existing deterministic frameworks for distributed systems, our extended small-world network provides good fault tolerance, since the links in the network are constructed probabilistically and are less vulnerable to adversarial attacks.

Chapter 5

Optimal Routing in a Small-World Network

In Chapter 4, we have proposed a one-dimensional extended small-world model with augmented local links, and presented both non-oblivious and oblivious routing algorithms that can route messages between any two nodes in $O(\lg n \lg \lg n)$ expected number of hops. In this chapter, we will propose a d -dimensional extended version of Kleinberg's small-world model, where each node has two more random links to nodes within certain Manhattan distance. Based on this model, we will present an oblivious decentralized algorithm that can finish routing in $O(\lg n)$ expected number of hops, which is an optimal routing complexity.

We extend Kleinberg's small-world model in that each node x has two more random links to nodes chosen uniformly and randomly within $(\lg n)^{\frac{2}{d}}$ Manhattan distance from x , where d denotes the dimension of the model. Based on this extended model, we then propose an oblivious algorithm that can route messages between any two nodes in $O(\lg n)$ expected number of hops, which is an optimal expected bound for routing. Our routing algorithm keeps only $O((\lg n)^{\beta+1})$ bits of information on each node, where $1 < \beta < 2$, thus being scalable with the network size. To our knowledge, our result is the first to achieve the optimal routing complexity while still keeping a poly-logarithmic number of bits of information stored on each node in the small-world networks.

The content of this chapter is organized as follows. Section 5.1 introduces the extended

small-world model and the decentralized routing algorithm. Section 5.2 lists our main results and contributions. In Section 5.3, we theoretically analyze our decentralized routing algorithm. Section 5.4 gives a brief conclusion of this chapter.

5.1 The Small-World Model and Decentralized Routing

Our small-world model is an extension of Kleinberg's d -dimensional model [61]. It is based on a d -dimensional torus $[n]_d = \{0, 1, \dots, n\}^d$ with three extra links for each node, where $d \geq 2$. Firstly, as in Kleinberg's original model [61], each node has a long-range link to another node chosen randomly according to the d -harmonic distribution, that is, the probability that node u sends a long-range link to another node v is $\Pr[u \rightarrow v] = \frac{1}{Z_u \cdot \text{Dist}(u, v)^d}$, where $\text{Dist}(u, v)$ denotes the Manhattan distance between nodes u and v , and $Z_u = \sum_{z \neq u} \frac{1}{\text{Dist}(u, z)^d}$. To avoid confusing with the extra links to be introduced shortly, we refer to such long-range links as the **K-type links** or **K-links** for short (where K stands for Kleinberg), and refer to node v as a **K-neighbor** of node u if there exists a K-link from u to v . Here we will introduce two more extra links for each node u to nodes that are chosen uniformly at random from nodes within $(\lg n)^{2/d}$ Manhattan distance from u . We refer to these two links as **augmented local links** or **AL-links** for short, and refer to node v as an **AL-neighbor** of node u if there exists an AL-link from u to v . Finally, we refer to the local links on the torus as **torus-links** or **T-links** for short, and refer to the local neighbors of node u on the torus as u 's **T-neighbors**. We refer to all nodes linked by u , including its K-neighbor, AL-neighbor and T-neighbor, as the *immediate neighbors* of node u .

We assume that all T-links on the torus are undirected, while all extra links including K-links and AL-links are directed. Obviously, there are $2d + 3$ immediate neighbors for each node in our extended small-world model. Thus, our extended model retains the same

CHAPTER 5. OPTIMAL ROUTING IN A SMALL-WORLD NETWORK

$O(1)$ order of node degree as that in Kleinberg's small-world model. Throughout this chapter, we use the terms *model* and *network* interchangeably.

In our decentralized routing algorithm, the message holder is also referred to as the *current node*. Given the current node x , let $\Gamma_x(0) = \{x\}$, and let $\Gamma_x(1)$ denote the AL neighborhood of all nodes in $\Gamma_x(0)$, and $\Gamma_x(2)$ denote the AL neighborhood of all nodes in $\Gamma_x(1)$, and so on. In other words, we refer to $\Gamma_x(i)$ as the i th level of AL neighborhood for node x , and refer to $A_x(i) = \bigcup_{j \leq i} \Gamma_x(j)$ as the first i levels of AL neighborhood for node x . At a certain level i of AL neighborhood, we may also refer to $A_x(i-1)$ as the set of *previously known nodes*. Let $L_x(i) = A_x(i) - A_x(i-1)$ denote the set of *new* nodes discovered during the i th level of AL neighborhood. We will call $A_x(k)$ the *AL awareness* of node x , where each node in our extended small-world model is aware of the first k levels of its AL neighborhood.

Recall that there are normally two approaches for decentralized routing: oblivious and non-oblivious schemes [39]. In an oblivious routing algorithm, each message holder makes routing decisions only based on its own routing table and the target node, while in a non-oblivious routing algorithm, each message holder depends on the previous routing history to make routing decisions. The scheme in [39] is oblivious, while the schemes in [65] and [72] are non-oblivious. In this chapter, we only consider the oblivious routing scheme.

The description of our oblivious routing algorithm is given as follows (Algorithm 1).

5.2 Our Contributions

Our main result is given as follows.

Theorem 5.1 *If each node in the extended small-world model is aware of the first $\beta \lg \lg n$ levels of its AL neighborhood, where $1 < \beta < 2$, then there exists an oblivious algorithm that can route messages between any two nodes in $O(\lg n)$ expected number of hops.*

CHAPTER 5. OPTIMAL ROUTING IN A SMALL-WORLD NETWORK

Algorithm 1**Input:** the source s and the target t .**Initialization:** $x \leftarrow s$.**The first phase:** $\text{Dist}(x, t) \geq (\lg n)^{\frac{2}{d}+1}$.

```

1: while  $\text{Dist}(x, t) \geq (\lg n)^{\frac{2}{d}+1}$  do
2:   node  $x$  checks in its AL awareness  $A_x(\beta \lg \lg n)$  whether there exists a node  $z$  that contains a K-neighbor within
      $m / \lg^\tau n$  Manhattan distance from  $t$ , where  $1 < \beta < 2$  and  $\tau$  denotes a certain constant which will be specified later.
     Let  $Z_x$  denote the set of such nodes  $z$ .
3:   if  $Z_x$  is empty then
4:     The message is routed to an immediate neighbor closest to  $t$ .
5:   else
6:     node  $x$  finds a node  $z$  in  $Z_x$  that is closest to  $x$  in terms of AL-links (ties are broken arbitrarily).
7:     node  $x$  computes a shortest path  $\pi : x_0 = x, x_1, \dots, x_t = z$  from  $x$  to  $z$  among  $A_x(\beta \lg \lg n)$ .
8:   end if
9:   if the shortest path  $\pi$  consists of only node  $x$  itself then
10:    The message is routed to node  $x$ 's K-neighbor.
11:  else
12:    node  $x$  routes the message to its next AL-neighbor  $x_1$  along the shortest path  $\pi$ .
13:  end if
14: end while

```

The final phase: $\text{Dist}(x, t) < (\lg n)^{\frac{2}{d}+1}$ The message is forwarded to an immediate neighbor closest to the target node t , until it reaches t .

Since any graph with $O(1)$ node degree has $\Omega(\lg n)$ expected path length, the $O(\lg n)$ expected bound of routing complexity in Theorem 5.1 is optimal. In addition, since the size of neighborhood $|A_x(\beta \lg \lg n)| = 1 + 2 + \dots + 2^{\beta \lg \lg n} < 2 \lg^\beta n$, and the address of each node is represented into a string of $O(\lg n)$ bits, the number of bits required on each node is at most $O((\lg n)^{\beta+1})$, where $1 < \beta < 2$. To our knowledge, this is the first result that achieves the optimal routing complexity while still keeping a poly-logarithmic number of bits stored on each node in the small-world networks. A comparison of our scheme with the other existing results is shown in Table 1. In the first three schemes (in [61, 8, 70, 71, 65]), we suppose that each node has q K-links, while in the next four schemes (in [72, 39, 101] and this chapter), we suppose that each node has one K-link.

Our results may be applied to the design of the logical overlay structure of large-scale distributed systems, such as peer-to-peer networks, in the same spirit as Symphony [70]. Since the links in the model are randomized, our extended network is less vulnerable to adversarial attacks, and thus provides good fault tolerance.

CHAPTER 5. OPTIMAL ROUTING IN A SMALL-WORLD NETWORK

Scheme	#bits of awareness on each node	#hops expected	Oblivious or Non-oblivious?
Kleinberg's greedy [61, 8, 11]	$O(q \lg n)$	$O(\lg^2 n / q)$	Oblivious
NoN-greedy [71]	$O(q^2 \lg n)$	$O(\lg^2 n / (q \lg q))$	Non-oblivious
Decentralized algorithm in [65]	$O(\lg^2 n / \lg(1+q))$	$O((\lg n)^2 / \lg^2(1+q))$	Non-oblivious
Decentralized algorithm [72]	$O(\lg^2 n)$	$O((\lg n)^{1+1/d})$	Non-oblivious
Indirect-greedy algorithm [39]	$O(\lg^2 n)$	$O((\lg n)^{1+1/d})$	Oblivious
Near-optimal algorithm for one-dimensional model with augmented awareness [Chapter 4]	$O(\lg^2 n)$	$O(\lg n \lg \lg n)$	Both are considered
Optimal algorithm for d -dimensional model with augmented awareness [this chapter]	$O((\lg n)^{\beta+1})$ ($1 < \beta < 2$)	$O(\lg n)$	Oblivious

Table 5.1: Comparisons of our decentralized routing algorithms with the other existing schemes.

5.3 Analysis of Decentralized Routing

In this section, we will give the proof of Theorem 5.1. A road map of the proof is given as follows. In Lemma 5.2, we first show that if each node is aware of the first $\beta \lg \lg n$ levels of its AL neighborhood, where $1 < \beta < 2$, then it is aware of at least $(\lg n)^{\delta+1}$ different nodes with certain probability, where $0 < \delta < \beta - 1$. Based on this result, in Lemma 5.3 we show that the AL awareness of each node is very likely to contain a K-neighbor that is close to the target node. Then in Lemma 5.7 and 5.9 we show that our oblivious routing algorithm can reduce the Manhattan distance effectively so that the $O(\lg n)$ expected bound of routing complexity can be achieved.

The following lemma from [8, 61] is useful for our subsequent analysis.

Lemma 5.1 *Let $\Pr[u \xrightarrow{K} v]$ denote the probability that node u sends a K -link to node v in a d -dimensional small-world model. Suppose that $a \leq \text{Dist}(u, v) \leq b$, then $\frac{c_1}{b^d \lg n} \leq \Pr[u \xrightarrow{K} v] \leq \frac{c_2}{a^d \lg n}$, where c_1 and c_2 are constants independent of n .*

We first quantify the size of the AL awareness of each node.

Lemma 5.2 *Let β denote a constant such that $1 < \beta < 2$. Let $A_x(\beta \lg \lg n)$ denote the AL awareness of node x in the extended small-world model, where each node is aware of the first $\beta \lg \lg n$ levels of its AL neighborhood. Then there exists a constant $0 < \delta < \beta - 1$*

CHAPTER 5. OPTIMAL ROUTING IN A SMALL-WORLD NETWORK

such that

$$\Pr[|A_x(\beta \lg \lg n)| \geq (\lg n)^{1+\delta}] > 1 - \frac{1}{\lg^\xi n},$$

where $\xi > 0$.

Proof: The proof is divided into two parts. In Part 1, we show that with probability at least $1 - 8(\lg n)^{-\frac{4}{3}}$, $|L_x(\frac{1}{3} \lg \lg n)| \geq (\lg n)^{\frac{1}{3}}$, that is, every AL-link during the first $\frac{1}{3} \lg \lg n$ levels of x 's AL neighborhood points to a new node. In Part 2, by using Chernoff's bound, we show that $|L_x(i)|$ still increases at an exponential rate for all $\frac{1}{3} \lg \lg n < i < \beta \lg \lg n$, based on which we have $|L_x(\beta \lg \lg n)| \geq (\lg n)^{1+\delta}$ with probability at least $1 - \frac{1}{\lg^\xi n}$, where $\xi > 0$.

Part 1: We will show that at each level of AL neighborhood (among the first $\frac{1}{3} \lg \lg n$ levels of AL neighborhood), the probability that an AL-link points to a previously known node is so small that all AL-links tend to point to new nodes, and hence $|L_x(\frac{1}{3} \lg \lg n)| \geq (\lg n)^{1/3}$ w.h.p..

We first calculate the following upper bound for $|A_x(\frac{1}{3} \lg \lg n)|$.

$$|A_x(\frac{1}{3} \lg \lg n)| \leq 1 + 2 + 2^2 + \dots + 2^{\frac{1}{3} \lg \lg n} = 2(\lg n)^{1/3} - 1 < 2(\lg n)^{1/3}.$$

Thus, we have $|A_x(i)| \leq |A_x(\frac{1}{3} \lg \lg n)| < 2(\lg n)^{1/3}$ for all $0 \leq i \leq \frac{1}{3} \lg \lg n$. Consider the construction of an AL-link of node x . Since each AL-link is connected to a node chosen randomly and uniformly from $\lg^2 n$ closest local nodes, each AL-link points to a node within the Manhattan distance $\lg^{2/d} n$ with an equal probability $(\lg n)^{-2}$. Since $|A_x(i)| \leq 2(\lg n)^{1/3}$, there are no more than $2(\lg n)^{1/3}$ previously known nodes at each level of AL neighborhood. Hence, the probability that any AL-link is connected to a previously known node is at most $2(\lg n)^{1/3} \cdot (\lg n)^{-2} = 2(\lg n)^{-5/3}$. Thus, the probability that an AL-link points to a new node is at least $1 - 2(\lg n)^{-5/3}$. There are in total at most $2|A_x(\frac{1}{3} \lg \lg n)| \leq 4(\lg n)^{1/3}$ number of AL-links, so all AL-links point to new nodes with probability at least $(1 - 2(\lg n)^{-5/3})^{4(\lg n)^{1/3}} \geq 1 - 8(\lg n)^{-4/3}$ for sufficiently large

CHAPTER 5. OPTIMAL ROUTING IN A SMALL-WORLD NETWORK

n . Here we use the fact $(1+x)^a \geq 1+ax$ for $x > -1$ and $a \geq 1$. Thus, we have $\Pr[|L_x(\frac{1}{3} \lg \lg n)| \geq (\lg n)^{1/3}] \geq 1 - 8(\lg n)^{-4/3}$.

Part 2: Let $\mathcal{B}$ denote the event that $|L_x(\frac{1}{3} \lg \lg n)| \geq (\lg n)^{1/3}$. From Part 1, we have $\Pr[\mathcal{B}] \geq 1 - 8(\lg n)^{-4/3}$. Next, we will consider the sequence of $L_x(i)$ for $\frac{1}{3} \lg \lg n \leq i \leq \beta \lg \lg n$. We assume that $|A_x(i)| \leq (\lg n)^{1+\delta}$ for all $\frac{1}{3} \lg \lg n \leq i \leq \beta \lg \lg n$, otherwise the lemma holds true. Since each AL-link is connected to a node chosen uniformly and randomly from $\lg^2 n$ closest local nodes, and there are at most $(\lg n)^{1+\delta}$ previously known nodes at each level of AL neighborhood, each AL-link reveals a new node with probability at least $1 - (\lg n)^{-2} \cdot (\lg n)^{1+\delta} = 1 - (\lg n)^{\delta-1}$. Let X_i denote the sum of $2|L_x(i)|$ independent Bernoulli random variables each with expectation $1 - (\lg n)^{\delta-1}$. Then $|L_x(i+1)|$ stochastically dominates X_i for all $\frac{1}{3} \lg \lg n \leq i \leq \beta \lg \lg n$. By Chernoff's bound, there exists a constant $0 < \epsilon < 1$ such that

$$\Pr[|L_x(i+1)| \leq 2(1 - (\lg n)^{\delta-1})(1 - \epsilon) \cdot |L_x(i)|] \leq \exp(-\epsilon^2(1 - (\lg n)^{\delta-1})|L_x(i)|).$$

Let $\mathcal{E}_i$ denote the event that $|L_x(i+1)| \geq 2(1 - (\lg n)^{\delta-1})|L_x(i)|(1 - \epsilon)$, where $\frac{1}{3} \lg \lg n \leq i \leq \beta \lg \lg n$, then we have $\Pr[\mathcal{E}_i] \geq 1 - \exp(-\epsilon^2(1 - (\lg n)^{\delta-1})|L_x(i)|)$. Let $\mathcal{E}$ denote the occurrences of the consecutive successful events $\mathcal{B}, \mathcal{E}_{\frac{1}{3} \lg \lg n}, \mathcal{E}_{\frac{1}{3} \lg \lg n + 1}, \dots, \mathcal{E}_{\beta \lg \lg n}$, then for large n , we have

$$\Pr[\mathcal{E}] \geq (1 - 8(\lg n)^{-4/3})(1 - \exp(-\epsilon^2(1 - (\lg n)^{\delta-1})(\lg n)^{1/3}))^{\beta \lg \lg n} > 1 - \frac{1}{\lg^\xi n},$$

where $\xi > 0$. At the last step, we applied the fact that $(1+x)^a \geq 1+ax$ for $x > -1$ and $a \geq 1$.

When the event $\mathcal{E}$ occurs, we have

$$\begin{aligned} |L_x(\beta \lg \lg n)| &\geq |L_x(\frac{1}{3} \lg \lg n)| \cdot (2(1 - (\lg n)^{\delta-1})(1 - \epsilon))^{(\beta - \frac{1}{3}) \lg \lg n} \\ &\geq (\lg n)^{1/3} (2(1 - (\lg n)^{\delta-1})(1 - \epsilon))^{(\beta - \frac{1}{3}) \lg \lg n} \\ &= (\lg n)^{1/3} (\lg n)^{(\beta - \frac{1}{3}) \lg (2(1 - (\lg n)^{\delta-1})(1 - \epsilon))} \\ &= (\lg n)^{\frac{1}{3} + (\beta - \frac{1}{3}) \lg (2(1 - (\lg n)^{\delta-1})(1 - \epsilon))} \end{aligned}$$

CHAPTER 5. OPTIMAL ROUTING IN A SMALL-WORLD NETWORK

Given a constant $1 < \beta < 2$, we can always find suitable constants $0 < \delta < \beta - 1$, $0 < \epsilon < 1$ and n_0 such that for all $n > n_0$, $\frac{1}{3} + (\beta - \frac{1}{3}) \lg(2((1 - (\lg n)^{\delta-1})(1 - \epsilon))) \geq 1 + \delta$. Thus, there exists a constant $0 < \delta < \beta - 1$ such that $\Pr[|A_x(\beta \lg \lg n)| \geq (\lg n)^{1+\delta}] > \Pr[|L_x(\beta \lg \lg n)| \geq (\lg n)^{1+\delta}] > 1 - \frac{1}{\lg^\xi n}$, where $\xi > 0$. Therefore, the lemma is verified. ■

Next, we will show that the AL awareness of the current node is very likely to contain a K-neighbor within $m/\lg^\tau n$ Manhattan distance from the target node t , where τ denotes a certain constant.

Lemma 5.3 *Suppose that the Manhattan distance between the current node x and the target node t in the extended small-world model is $m \geq (\lg n)^{\frac{2}{d}+1}$. Then there exists a constant τ such that with probability at least $1 - \frac{1}{\lg^\zeta n}$, where $\zeta > 0$, x 's AL awareness $A_x(\beta \lg \lg n)$ contains a K-neighbor within $\frac{m}{\lg^\tau n}$ Manhattan distance from the target node t .*

Proof: Let $\mathcal{C}$ denote the event that $|A_x(\beta \lg \lg n)| \geq (\lg n)^{1+\delta}$. By Lemma 5.2, we have $\Pr[\mathcal{C}] > 1 - \frac{1}{\lg^\xi n}$, where $\xi > 0$.

Let $D_l(t)$ denote the set of all nodes within l Manhattan distance from t . Given a node u in $A_x(\beta \lg \lg n)$, let $\Pr[u \xrightarrow{K} D_l(t)]$ denote the probability that u 's K-neighbor is inside the ball $D_l(t)$.

Since each AL-link spans the Manhattan distance no more than $(\lg n)^{2/d}$, the nodes in x 's AL awareness $A_x(\beta \lg \lg n)$ are all within $\beta \lg \lg n (\lg n)^{2/d}$ Manhattan distance from x . Since $\text{Dist}(x, t) = m \geq (\lg n)^{\frac{2}{d}+1}$, the maximum Manhattan distance between a node in $A_x(\beta \lg \lg n)$ and any node in $D_{\frac{m}{\lg^\tau n}}(t)$ is no more than $(m + \frac{m}{\lg^\tau n} + \beta \lg \lg n (\lg n)^{2/d}) \leq 2m$. By Lemma 5.1, the probability for u 's K-neighbor to be inside the ball $D_{\frac{m}{\lg^\tau n}}(t)$ is at least $\frac{c_1}{(2m)^d \lg n}$, so we have

$$\Pr[u \xrightarrow{K} D_{\frac{m}{\lg^\tau n}}(t)] \geq |D_{\frac{m}{\lg^\tau n}}(t)| \cdot \frac{c_1}{(2m)^d \lg n} = \left(\frac{m}{\lg^\tau n}\right)^d \cdot \frac{c_1}{(2m)^d \lg n} = \frac{c_3}{(\lg n)^{\tau d+1}},$$

where $c_3 = c_1/2^d$ denotes a constant.

CHAPTER 5. OPTIMAL ROUTING IN A SMALL-WORLD NETWORK

Let $\Pr[A_x(\beta \lg \lg n) \xrightarrow{K} D_{\frac{m}{\lg^\tau n}}(t)]$ denote the probability that at least one node in $A_x(\beta \lg \lg n)$ contains a K -neighbor within $D_{\frac{m}{\lg^\tau n}}(t)$. Then if $\tau d < \delta$, we have

$$\begin{aligned}
 \Pr[A_x(\beta \lg \lg n) \xrightarrow{K} D_{\frac{m}{\lg^\tau n}}(t)] &\geq \Pr[A_x(\beta \lg \lg n) \xrightarrow{K} D_{\frac{m}{\lg^\tau n}}(t) \mid \mathcal{C}] \cdot \Pr[\mathcal{C}] \\
 &\geq \left(1 - \left(1 - \frac{c_3}{\lg^{\tau d+1} n}\right)^{(\lg n)^{1+\delta}}\right) \cdot \left(1 - \frac{1}{\lg^\zeta n}\right) \\
 &\geq \left(1 - \exp(-c_3(\lg n)^{\delta-\tau d})\right) \cdot \left(1 - \frac{1}{\lg^\zeta n}\right) \\
 &\quad (\text{using the fact } (1 + \frac{b}{x})^x \leq e^b, b \in \mathbb{R}, x > 0) \\
 &> 1 - \frac{1}{\lg^\zeta n} \quad (\text{because } \tau d < \delta),
 \end{aligned}$$

for a constant $\zeta > 0$. ■

By using a similar technique as that in Lemma 5.3, we can have the following lemma.

Lemma 5.4 *Suppose that the Manhattan distance between the current node x and the target node t in the extended small-world model is $m \geq (\lg n)^{\frac{2}{d}+1}$. Then there exists a constant τ such that with probability at least $1 - \frac{1}{\lg^\zeta n}$, where $\zeta > 0$, x 's AL awareness $A_x(\beta \lg \lg n)$ contains a K -neighbor within $\frac{m - \beta(\lg n)^{2/d} \lg \lg n}{\lg^\tau n}$ Manhattan distance from the target node t .*

Given the current node x and the target node t , we refer to the set of nodes within $\text{Dist}(x, t)/\lg^\tau n$ Manhattan distance from t as the *influenced set* of node x , where τ is a certain constant as given above.

Lemma 5.5 *Suppose that the Manhattan distance between the current node x and the target node t in the extended small-world model is $m \geq (\lg n)^{\frac{2}{d}+1}$. Then a node within $(m - \beta(\lg n)^{2/d} \lg \lg n)/\lg^\tau n$ Manhattan distance from t is also inside the influenced set of any node in $A_x(\beta \lg \lg n)$.*

CHAPTER 5. OPTIMAL ROUTING IN A SMALL-WORLD NETWORK

Proof: Since each AL-link spans no more than $(\lg n)^{2/d}$ Manhattan distance from the current node, and there are in total $\beta \lg \lg n$ levels of AL neighborhood for x , all nodes in $A_x(\beta \lg \lg n)$ span no more than $\beta(\lg n)^{2/d} \lg \lg n$ Manhattan distance from x . By this simple observation, the proof of the lemma can be easily obtained. ■

Lemma 5.6 *Suppose that the Manhattan distance between the current node x and the target node t in the extended small-world model is $m \geq (\lg n)^{\frac{2}{d}+1}$. Let I_1 denote the set of nodes within Manhattan distance $\frac{m}{\lg^\tau n}$ from t , and let I_2 denote the set of the nodes within Manhattan distance $\frac{m - \beta(\lg n)^{2/d} \lg \lg n}{\lg^\tau n}$ from t . Then the probability for $A_x(\beta \lg \lg n)$ to contain a K -neighbor within $I_1 - I_2$ is no more than $\frac{1}{\lg^\varphi n}$, where $\varphi > 0$.*

Proof: We first calculate an upper bound of $|I_1 - I_2|$. We have

$$|I_1 - I_2| = \left(\frac{m}{\lg^\tau n}\right)^d - \left(\frac{m - \beta(\lg n)^{2/d} \lg \lg n}{\lg^\tau n}\right)^d = \frac{m^d - (m - \beta(\lg n)^{2/d} \lg \lg n)^d}{\lg^{\tau d} n}.$$

When $m \geq (\lg n)^{2/d+1}$, we have $(m - \beta(\lg n)^{2/d} \lg \lg n)^d \geq m^d - m^{d-1}(\beta(\lg n)^{2/d} \lg \lg n)$.

Thus, we can obtain the following upper bound for $|I_1 - I_2|$

$$|I_1 - I_2| \leq \frac{m^{d-1}(\beta(\lg n)^{2/d} \lg \lg n)}{\lg^{\tau d} n}.$$

By Lemma 5.1, the probability for a node y_1 in $A_x(\beta \lg \lg n)$ to send a K -link to a node y_2 in $I_1 - I_2$ is at most $\frac{c_2}{(m/2)^d \lg n}$, since $\text{Dist}(y_1, y_2) \geq m/2$ if $m \geq (\lg n)^{\frac{2}{d}+1}$. Since $|A_x(\beta \lg \lg n)| \leq 2 \lg^\beta n$, we have

$$\begin{aligned} & \Pr[A_x(\beta \lg \lg n) \text{ contains a } K\text{-neighbor within } I_1 - I_2] \\ & \leq \frac{c_2}{(m/2)^d \lg n} \cdot |A_x(\beta \lg \lg n)| \cdot |I_1 - I_2| \\ & \leq \frac{2^d c_2}{m^d \lg n} \cdot 2 \lg^\beta n \cdot \frac{m^{d-1}(\beta(\lg n)^{2/d} \lg \lg n)}{\lg^{\tau d} n} \\ & = \frac{2 \cdot 2^d c_2 \beta (\lg n)^{\beta + \frac{2}{d}} \lg \lg n}{m (\lg n)^{1+\tau d}} \\ & \leq \frac{2 \cdot 2^d c_2 \beta (\lg n)^\beta \lg \lg n}{(\lg n)^{2+\tau d}} \quad (\text{since } m \geq (\lg n)^{\frac{2}{d}+1}) \\ & \leq \frac{1}{\lg^\varphi n} \quad (\text{where } \varphi > 0) \end{aligned}$$

CHAPTER 5. OPTIMAL ROUTING IN A SMALL-WORLD NETWORK

Thus, the lemma is verified. ■

Lemma 5.7 *Suppose that the Manhattan distance between the current node x and the target node t in the extended small-world model is $m \geq (\lg n)^{\frac{2}{d}+1}$. Then after at most $O(\lg \lg n)$ expected number of hops, the message will reach a node within $m/\lg^\tau n$ Manhattan distance from t , where τ denotes a certain constant.*

Proof: As in Lemma 5.6, let I_1 denote the set of nodes within Manhattan distance $\frac{m}{\lg^\tau n}$ from t , and let I_2 denote the set of the nodes within Manhattan distance $\frac{m - \beta(\lg n)^{2/d} \lg \lg n}{\lg^\tau n}$ from t . Let $\mathcal{F}_1$ denote the event that $A_x(\beta \lg \lg n)$ contains a K-neighbor within $I_1 - I_2$, and let $\mathcal{F}_2$ denote the event that $A_x(\beta \lg \lg n)$ contains a K-neighbor within I_2 . By Lemma 5.6 and Lemma 5.4, we have $\Pr[\mathcal{F}_1] \leq \frac{1}{\lg^\varphi n}$ and $\Pr[\mathcal{F}_2] \geq 1 - \frac{1}{\lg^\zeta n}$ respectively, where $\varphi, \zeta > 0$. Thus, with probability at least a positive constant, $A_x(\beta \lg \lg n)$ contains a K-neighbor within I_2 , but no K-neighbor within $I_1 - I_2$.

We refer to the routing steps from a given node x to its intermediate node z in $A_x(\log \log n)$ as an indirect phase. The routings in different indirect phases are independent from each other. By above statement, after at most $O(1)$ expected number of indirect phases, i.e., at most $c' \cdot \beta \lg \lg n$ expected number of hops for a constant c' , the message will be routed to a node x whose AL awareness contains a K-neighbor within I_2 , but no K-neighbor within $I_1 - I_2$. Let z be the intermediate node in $A_x(\beta \lg \lg n)$ that contains a K-neighbor within the influenced set I_1 (or I_2) and is closest to node x in terms of AL-links. Next, we will show that after the event $\mathcal{F}_2 \cap \neg \mathcal{F}_1$ occurs, our oblivious algorithm will route the message to the intermediate node z along a shortest path $\pi : x_0 = x, x_1, \dots, x_t = z$ among $A_x(\beta \lg \lg n)$.

We refer to a node z in $A_x(\lg \lg n)$ as a *good intermediate node* if it satisfies the following two conditions: (1) has a K-neighbor within $\frac{m}{\lg^\tau n}$ to the target node; (2) is closest to node x in terms of AL-links.

CHAPTER 5. OPTIMAL ROUTING IN A SMALL-WORLD NETWORK

We first consider the case where no tie arises, that is, x 's good intermediate node z is unique. By our oblivious algorithm, node x will route the message to its next AL-neighbor x_1 along a shortest path π . From Lemma 5.5, node z 's K-neighbor is also inside the influenced set of x_1 , and hence it satisfies the first condition of a good intermediate node for x_1 . In addition, since z is an intermediate node closest to x , it is also an intermediate node closest to x_1 . Thus, it also satisfies the second condition of a good intermediate node for x_1 . Therefore, node x_1 will also regard node z as its good intermediate node, and find a shortest path $\pi : x_1, x_2, \dots, x_t = z$ from x_1 to z , and then route the message to its next AL-neighbor x_2 . Such a process is repeated for every node x_i on the shortest path π until the message reaches the intermediate node z . After that, the message will be routed to z 's K-neighbor and hence reach a node within $m/\lg^\tau n$ Manhattan distance from the target node t .

When the ties happens, there may be more than one good intermediate nodes z for the current node x . However, it is easy to show that the message will be routed to a good intermediate node z along one of shortest paths. This does not affect the result. Therefore, the proof of the lemma is completed. ■

Lemma 5.8 *Suppose that the Manhattan distance between the current node x and the target node t in the extended small-world model is $m \geq (\lg n)^{\frac{2}{d}+1}$. Then after at most $O(\lg n)$ expected number of hops, our oblivious routing algorithm can reduce the Manhattan distance to within $(\lg n)^{\frac{2}{d}+1}$.*

Proof: By Lemma 5.7, after at most $O(\lg \lg n)$ expected number of hops, the message will reach a node within $\frac{m}{\lg^\tau n}$ Manhattan distance from t , where τ denotes a certain constant.

Divide the whole Manhattan distance $\text{Dist}(x, t)$ into phases such that the i th phase contains the nodes within $[\frac{m}{(\lg n)^{\tau i}}, \frac{m}{(\lg n)^{\tau(i-1)}})$ Manhattan distance from t . Since the maximum Manhattan distance is n , there are at most $O(\frac{\lg n}{\lg \lg n})$ phases. Because each phase takes at

CHAPTER 5. OPTIMAL ROUTING IN A SMALL-WORLD NETWORK

most $O(\lg \lg n)$ expected number of steps by Lemma 5.7, after $O(\frac{\lg n}{\lg \lg n}) \cdot O(\lg \lg n) = O(\lg n)$ hops, the Manhattan distance can be reduced to within $(\lg n)^{\frac{2}{d}+1}$. ■

Lemma 5.9 *Suppose that the Manhattan distance between the current node x and the target node t in the extended small-world model is $m < (\lg n)^{\frac{2}{d}+1}$. Then using Kleinberg's greedy algorithm can route the message to the target node t in $O(\lg n)$ expected number of hops.*

Proof: When $\text{Dist}(x, t) < (\lg n)^{\frac{2}{d}+1}$, Kleinberg's greedy algorithm is executed, that is, the message is forwarded to an immediate neighbor closest to t .

Since each AL-neighbor is chosen uniformly at random from nodes within $(\lg n)^{2/d}$ Manhattan distance from x , the probability for an AL-link to jump over $(\lg n)^{2/d}/2$ Manhattan distance is $1 - 2^{-d}$. Thus, after at most $O(1)$ expected number of hops, the Manhattan distance will be reduced by $(\lg n)^{2/d}/2$. Therefore, after at most $O(\lg n)$ expected number of hops, the Manhattan distance will be reduced to within $(\lg n)^{2/d}$. After that, using the simple greedy algorithm via the local T-links can route the message to the target node t in $(\lg n)^{2/d}$ hops. Since $d \geq 2$, $(\lg n)^{2/d} = O(\lg n)$. Therefore, after in total $O(\lg n) + (\lg n)^{2/d} = O(\lg n)$ expected number of hops, the message will reach the target node t . ■

Combining Lemma 5.7 and Lemma 5.9 together, we obtain the proof of Theorem 5.1.

5.4 Concluding Remarks

We have extended Kleinberg's small-world network with two more augmented local links, and shown that if each node in the network is aware of $\beta \lg \lg n$ levels of augmented local neighborhood, where $1 < \beta < 2$, there exists an oblivious decentralized algorithm that can finish routing in $O(\lg n)$ expected number of hops, which is an optimal expected bound for routing.

CHAPTER 5. OPTIMAL ROUTING IN A SMALL-WORLD NETWORK

In this chapter, we only focus on the oblivious routing scheme. A non-oblivious algorithm can be easily obtained based on the design and analysis of our oblivious scheme.

Chapter 6

Experimental Study of Small-World Networks and Applications to Peer-to-Peer Systems

In this chapter, we experimentally study the routing issues in the current small-world models, including existing routing schemes in Kleinberg's models, our two versions of routing schemes for the extensions of Kleinberg's models, and the greedy routing in Watts-Strogatz's models. We then investigate certain issues about deploying small-world overlay structures for peer-to-peer systems, such as the improvement of routing in Symphony [70], a well-known peer-to-peer overlay network with small-world-like connections, and the estimation of the global network size even under non-uniform node distribution in the identifier space.

The content of this chapter is organized as follows. In Section 6.1, we experimentally verify our results in Chapter 4 and Chapter 5. In Section 6.2, we then apply these results to improve the performance of Symphony. Section 6.3 presents the routing experiments in Watts-Strogatz's small-world network, a well-known model of the small-world phenomenon in social networks. In Section 6.4, we explore some sampling methods for estimating the network size for networks with non-uniform node distribution. Section 6.5 gives the concluding remarks.

6.1 Experimental Study of Routing in Kleinberg's and Extended Small-World Networks

In Chapter 4 and Chapter 5, we have proposed extended versions of Kleinberg's small-world networks, based on which we have presented near-optimal and optimal algorithms for routing messages between any two nodes. In this section, we will conduct experiments to evaluate our schemes and other existing routing schemes for Kleinberg's small-world networks.

We will focus on the following five schemes: (a) The original greedy routing algorithm [61] in Kleinberg's small-world network with only one long-range contact per node, where each node forwards the message to its immediate neighbor closest to the destination; (b) The greedy routing algorithm in Kleinberg's small-world network with two long-range contacts per node [8, 70]. In the experimental study, we would like to learn how much the additional number of long-range links can help routing; (c) The decentralized routing scheme with $O(\lg n)$ local awareness [39, 72]. With this scheme, we intend to evaluate the degree at which the local awareness improves routing efficiency; (d) Our near-optimal routing scheme in Chapter 4; (e) Our optimal routing scheme in Chapter 5.

We note that most schemes have non-oblivious and oblivious versions. For example, both non-oblivious [72] and oblivious [39] algorithms with $O(\lg n)$ local awareness can achieve $(\lg n)^{1+1/d}$ routing complexity in a d -dimensional model, where n denotes the number of nodes in the network. Here we only focus on the non-oblivious version for each scheme.

6.1.1 Experimental Setup

Network Construction: We construct the small-world network based on a 2-D grid $[1, \dots, \sqrt{n}] \times [1, \dots, \sqrt{n}]$. Let $N(i, j)$ denote a node with the row coordinate i and the column coordinate j . Each node $N(i, j)$ is connected to its immediate neighbors $N((i$

CHAPTER 6. EXPERIMENTAL STUDY OF SMALL-WORLD NETWORKS AND APPLICATIONS TO PEER-TO-PEER SYSTEMS

$\text{mod } n) \mp 1, (j \text{ mod } n) \mp 1$). Let $H_n = \sum_{i=1}^n 1/i$ denote the harmonic normalization factor. We then generate a sequence of intervals $I(i)$, which we call the probability intervals, where $1 \leq i \leq 2\sqrt{n} - 1$. Let $0 < I_1 \leq 1/H_n$, and $\frac{1}{(i-1)H_n} < I_i \leq \frac{1}{iH_n}$, where $2 \leq i \leq 2\sqrt{n} - 1$. Each node $N(i, j)$ uniformly generates a random number x in $(0, 1]$, and then finds the interval that contains x . Suppose that x is located in the interval I_k , node $N(i, j)$ then forms a long-range link connecting to a node at Manhattan distance k (If there are more than one such node, one is randomly chosen). When each node has multiple long-range contacts, it just generates more than one random numbers, and sets up the connections in the same way.

In the extension of Kleinberg's small-world networks, each node uniformly and randomly chooses two nodes within the Manhattan distance $\lg^2 n$ as its augmented local neighbors.

Messages Generation and Evaluation Metrics: We let each node generate a query message with a random destination, and then evaluate the following metrics.

- (1) **Average length of routing path** is the average number of hops traversed by the messages from the source to the destination.
- (2) **Storage requirement** for each node is the number of bits required to be stored on each node.

6.1.2 Experimental Results

We vary the number of nodes in the network from 5,000 to 25,000, and evaluate different routing schemes, as shown in Figures 6.1 and 6.2. For large n , the greedy algorithm with increasing number of long-range contacts [8, 70], the decentralized routing algorithm with local awareness [39, 72], our near-optimal and optimal algorithms all improve Kleinberg's original greedy algorithm. Our optimal algorithm performs the best and it reduces the latency (in terms of number of hops) by around 40%, but at the cost of increased amount of information stored on each node. On the other hand, our near-optimal scheme can find a

CHAPTER 6. EXPERIMENTAL STUDY OF SMALL-WORLD NETWORKS AND APPLICATIONS TO PEER-TO-PEER SYSTEMS

shorter routing path than the decentralized routing schemes with local awareness [39, 72], while keeping the same storage space on each node.

From Figure 6.1 and Figure 6.2, we can see that both the average routing path and the storage requirement for each node in all schemes increase moderately as the network size increases, thus all of them are scalable.

Figure 6.1: Average length of routing paths for different routing schemes.

6.2 Improved Symphony with Augmented Local Awareness

Symphony [70] is a peer-to-peer overlay structure that applies Kleinberg's small-world model to connect nodes in the dynamic network. In this section, we enhance Symphony's structure by adding augmented local links to each node. We refer to the improved structure as Symphony*. In Symphony*, we use the near-optimal algorithm in Chapter 4 to route messages.

6.2.1 Construction of Symphony*

As in [70], we let both data indexes and node addresses hash into the ring space $(0, 1]$. When a node joins the network, it selects a random point uniformly in the space $(0, 1]$ as insertion

CHAPTER 6. EXPERIMENTAL STUDY OF SMALL-WORLD NETWORKS AND APPLICATIONS TO PEER-TO-PEER SYSTEMS

Figure 6.2: Storage requirement on each node for different routing schemes.

position, and then maintains the data indexes located between it and its predecessor. The connections of each node are constructed as follows. Firstly, each node has local ring links connected to its predecessor and successor along the ring. Secondly, each node forms a long-range link as in [70], where the long-range neighbor is chosen according to the probability distribution function (pdf) $p_n(x) = 1/(x \cdot \lg n)$, where $x \in [1/n, 1]$, and n is the estimated number of nodes in the network based on local node density. In addition, each node at point u has two primary augmented local links (or PAL-links for short) connected to nodes chosen uniformly at random within the interval $(u, u + \frac{\lg^2 n}{n}]$, and a secondary augmented local link (or SAL-link for short) connected to a node chosen uniformly at random within the interval $(u, u + \frac{\lg n}{n}]$.

As in Chapter 4, each node in Symphony* is aware of $\lg \lg n$ levels of primary augmented local neighborhood. During the routing, each node forwards the message by using an algorithm similar to the near-optimal algorithm in Chapter 4.

CHAPTER 6. EXPERIMENTAL STUDY OF SMALL-WORLD NETWORKS AND APPLICATIONS TO PEER-TO-PEER SYSTEMS

6.2.2 Evaluation of Routing Performance

We evaluate the routing performance of Symphony* in which each node generates a message with a random destination. We compare the routing path length (on average) and the cost of node joining and departure for the original Symphony and Symphony*.

Figure 6.3 evaluates the length of routing path in the original and improved Symphony. We see that the length of routing path in Symphony* is shorter than that of the original Symphony. As n increases, the improvement of routing latency becomes more obvious, which is consistent with the theoretical observation in Chapter 4.

Figure 6.3: Average length of routing path in the original Symphony and Symphony*

Figure 6.4 plots the cost of node joining/leaving, which is measured by the number of messages required to be sent for maintaining the DHT properties. Figure 6.4 shows that the number of required messages is decreased in Symphony* when n is large. Thus, Symphony* supports better routing efficiency while keeping lower maintenance cost compared with the original Symphony.

6.2.3 Evaluation of Resilience to Link Failure

Figure 6.5 compares the fault tolerance of the original Symphony and Symphony*, given that each link or node fails with some probability. To guarantee the success of each search,

CHAPTER 6. EXPERIMENTAL STUDY OF SMALL-WORLD NETWORKS AND APPLICATIONS TO PEER-TO-PEER SYSTEMS

Figure 6.4: Cost of node joining/leaving in the original Symphony and Symphony*

we assume that each node is always connected to its predecessor and successor. The long-range and augmented local links fail with probability p , whereas remain useable with probability $1 - p$. We then evaluate the change of routing latency with respect to varying link failure probability p . Figure 6.5 shows that Symphony* keeps more routing efficiency than the original model when the links in both of them have the same failure probability.

Figure 6.5: Comparison of fault tolerance between Symphony and Symphony*

6.3 Routing in Watts-Strogatz's Small-World Networks

6.3.1 Watts-Strogatz's Small-World Network

In [27], Watts and Strogatz introduce the notation of “small-world” networks to describe networks with low diameter, which generally measures a maximum of hop distance between any two nodes, and high clustering coefficient, as compared with traditional random graphs. They give a rewiring procedure to interpolate between regular and random connections. Their model starts with an n -node ring with k links to closest neighbors along the ring for each node. Then with probability p , each link is reconnected to another node chosen uniformly at random over the entire ring, whereas with probability $1 - p$, each link remains unchanged.

Based on this rewiring model, Watts and Strogatz experimentally study the diameter $D(p)$ and the clustering coefficient $C(p)$ of the network when varying the value of p . They find that there is a broad range of p when $D(p)$ is almost as small as that of the traditional random graphs, while $C(p)$ is much larger than that of conventional random graphs, which will be also shown in our subsequent experimental study.

Although Watts-Strogatz networks exhibit small graph diameters and high clustering coefficients, it remains unknown whether there exists an efficient routing path between any two nodes in their network. The diameter measures the separation between any two nodes in the graph, but as pointed out by Kleinberg in [61], a graph with a low diameter may not always admit an efficient routing algorithm. Meanwhile, the clustering coefficient measures the “cliquishness” of the neighborhood for each node in the graph, but the relation between the routing path and the clustering coefficient remains unknown. It is not clear whether the high clustering coefficient will result in a short routing path. As we will observe in our experiment, this is not always the case. Our study aims to provide short routing paths in the small-world networks.

6.3.2 Experimental Setup

In Watts-Strogatz's network, all edges are undirected. In order to study the routing issue, we consider a directed version of Watts-Strogatz's network, in which each edge is directed. In order to guarantee connectivity of the graph and hence the success of routing, we let each node in the ring always keep the connection to its successor, and only rewire the other $k - 1$ links. Such a modification allows us to study the impact of the value of k while still keeping the entire network connected.

6.3.3 Performance Metrics

Besides the diameter and the clustering coefficient, we also evaluate the length of routing path in our simulation.

- (1) **Average distance.** The average distance is the average shortest path over all pairs of nodes in the network. We will let $D(p)$ denote the average distance of Watts-Strogatz's network with the rewiring probability p .
- (2) **Average clustering coefficient.** The clustering coefficient of a node u is the ratio between the number of edges and the maximum number of edges allowed among u 's neighbors. The average clustering coefficient of a graph is the average clustering coefficient over all nodes in the graph. Let $C(p)$ denote the clustering coefficient of Watts-Strogatz's network with rewiring probability p .
- (3) **Average length of routing path.** The length of a routing path is the number of hops from the source to the destination using the decentralized routing algorithm, in which each node participating in routing uses only its local information and routes the message greedily to its neighbor closest to the destination (we assume that each node is aware of the underlying ring structure and knows its own and the destination's location on the ring). The average length of routing path is the average over all

routing paths. Let $R(p)$ denote the average routing length of routing path for the Watts-Strogatz's network with rewiring probability p .

Let $D(0)$, $C(0)$ and $R(0)$ represent the average distance, clustering coefficient and length of routing path for Watts-Strogatz's network with rewiring probability 0, that is, the initial regular graph without edge rewiring. We evaluate the above three metrics normalized by $D(0)$, $C(0)$ and $R(0)$ respectively, and plot their curves for different values of p .

6.3.4 Experimental Results

We first examine the curves of $D(p)$, $C(p)$ and $R(p)$ for different n ($n = 500, 800, 1000$). As in Watts-Strogatz's experiments, we choose $k \approx \lg n$. Figure 6.6 illustrates the curves of $D(p)/D(0)$, $C(p)/C(0)$ and $R(p)/R(0)$ for different n . As observed in [27], there is a rapid drop in $D(p)$ as the rewiring probability p increases, while $C(p)$ remains almost the same as its initial value $C(0)$ for the regular graph. Watts and Strogatz believe the small-world phenomenon begins to emerge at this range of p . However, we see that although there is also a rapid drop in $R(p)$, efficient routing does not emerge along with small diameters. In other words, the low diameter and the high clustering coefficient may not imply the existence of efficient routing path. On the other hand, we find that both $D(p)$ and $c(p)$ decrease as p increases, whereas $R(p)$ first rapidly declines, and then smoothly decreases for a large range of p , but increases again as p tends to 1. This suggests that greedy routing is not efficient in a completely random graph. As theoretically proved in Kleinberg's result [61], the complete random graph with n nodes required at least $\Omega(\sqrt{n})$ number of hops for routing messages.

We also note that the most efficient (greedy) routing occurs around $p = 0.6$, where the shortest routing path is about 2 times of the diameter of the network. This suggests that the best Watts-Strogatz's small-world overlay for peer-to-peer systems has a re-wiring probability of about $p = 0.6$.

We next evaluate the three metrics for different k values as shown in Figure 6.7. We choose $n = 200$ in our simulation. From Figures 6.7(a) and 6.7(e), we see that the rewiring models with small k ($k = 2$) and with large k ($k = 20$) “shrink” the initial regular graph less efficiently. For smaller values of k , it is because that the number of rewired edges is so small that the rewiring procedure cannot “tighten” the graph efficiently. For larger values of k , the diameter of the initial graph is already very small.

6.4 Network Size Estimation for Non-Uniform Node Distribution

Most existing structured peer-to-peer applications use distributed hash tables (DHTs) to construct the logical overlay networks, e.g. Chord [97], CAN [87], where both data items and node addresses are uniformly hashed into a same naming space, and the query message is forwarded according to the distance metric in the naming space. Symphony [70] is an example of DHT which incorporates Kleinberg’s small-world graph to connect nodes uniformly distributed in a ring space. In Kleinberg’s small-world networks, each node needs the global network size n to obtain the normalizing factor to calculate the probability for setting up the long-range connection. Symphony applies a technique based on local densities, as used in [49, 69], to estimate the network size n . Since hashed node identifiers¹ are uniformly and randomly distributed along the ring space, it is not surprising that the local density of each node reflects the average node density over the entire naming space.

DHTs support rare item queries, which is an advantage over other unstructured peer to peer systems, such as Kazza [59], Gnutella [42], which can only support popular content searching. However, DHTs have poor performance in partial-match searches. In order to preserve the semantic proximity of data items, nodes have to be mapped to points semantically clustered in some domain or space, such as Latent Semantic Indexing (LSI)

¹The terms node and node identifier are used interchangeably throughout this chapter.

space [67, 99]. Nevertheless, this means that nodes cannot be hashed uniformly into some metric space like DHTs, which will make the task of estimating of the network size more difficult.

In the following section, we propose two sampling schemes to estimate the global network size for non-uniform node distribution in the construction of overlay networks. To our knowledge, we are the first to explore this issue. Although in [67], Li et al. propose a semantic small-world overlay network to facilitate efficient semantics based search, it is not clear how their small-world connections can be set up without knowing the global network size for each individual node. We believe that our work can be combined with the results in [67] to support partial-match search.

6.4.1 Our Sampling Algorithms

We propose two sampling approaches to estimate the network size, and compare their performance in the experimental study. The first approach (Algorithm 1) probes the space uniformly and gets a sample set of nodes, and then calculates their average density as the estimated value of the network size. The second one (Algorithm 2) consists of two stages: In the first stage, each node uniformly probes the space and gets a sample set of nodes. After that, a second-stage probe is executed for each node in the sample set within a local window based on its local density, that is, a node that found a higher local density in the first-stage will sample more in its neighborhood. At the end, the average density over all sampled nodes is regarded as the estimated value of the network size.

It may appear that the second approach would have better estimation accuracy than the first one. However, our experimental results show that is not true, and indeed the first approach with simple uniform sampling has a desirable outcome. We assume that nodes are mapped into a ring space $(0, 1]$ according to some distribution. The cases for multiple dimensional space can be easily extended based our results.

CHAPTER 6. EXPERIMENTAL STUDY OF SMALL-WORLD NETWORKS AND APPLICATIONS TO
PEER-TO-PEER SYSTEMS

Algorithm 1: Uniform Sampling Algorithm

1. Each node uniformly and randomly probes M_1 number of sample nodes.
 2. The local density of each node in the sample set is calculated according to the following methods: Let d_i denote the ring segment spanned by L nodes closest to the sampling node i . Then the local density is equal to L/d_i .
 3. Compute the average local density over all sampling nodes: $N_{estimate} = (\sum_{i=1}^{M_1} \frac{L}{d_i})/M_1$.
-

Algorithm 2: Density Based Sampling Algorithm

1. Uniformly probe M_2 nodes in the space as the first-stage sample set.
 2. Calculate the local density of each node in the first-stage sample set according to the following method: Let d_i denote the ring segment spanned by L nodes closest to the sampling node i . Then the local density is equal to L/d_i .
 3. Second-stage sampling: For each node in the first-stage sample set, probe $\alpha \cdot \frac{L}{d_i}$ nodes more, uniformly within the distance w as the second-stage sample set, where α denotes the zooming factor, and w denotes the local window size for second-stage sampling.
 4. Calculate the average local density over all nodes in the second-stage sample set as the estimated value of the global network size.
-

6.4.2 Data Generation for Testing

We generate four data patterns of node distribution for evaluating our sampling approaches:

- (1) Uniform distribution. As in Symphony [70], nodes are uniformly distributed in the ring space.
- (2) Uniform-cluster distribution. There are m clusters in the ring, and the cluster sizes follow the uniform distribution. To generate the data, we first randomly generate m random points as the centers of the clusters. Then each node is assigned to a cluster

CHAPTER 6. EXPERIMENTAL STUDY OF SMALL-WORLD NETWORKS AND APPLICATIONS TO PEER-TO-PEER SYSTEMS

with equal probability. Within each cluster, nodes follow a Gaussian distribution with the mean at the cluster center, and the standard deviation $\sigma = 0.05$.

- (3) Zipf-cluster distribution. There are m clusters each with center at a random point. The cluster sizes follow a Zipf distribution, that is, a node is located within the cluster i with probability $p_i = 1/(H_m \cdot i)$, where $H_m = \sum_{j=1}^m 1/j$, denotes the m -th harmonic number. Within each cluster, nodes are distributed according to a Gaussian distribution with the mean at the cluster center, and the standard deviation $\sigma = 0.05$.
- (4) One-cluster distribution. Most nodes are within one cluster with a random point as the center. All nodes in the cluster follow a Gaussian distribution with the mean at the cluster center and the standard deviation $\sigma = 0.05$.

Figure 6.8 shows examples of these four distribution patterns.

6.4.3 Experimental Setup

Our experimental parameters, including their values and the defaults, are given in Table 6.1.

Symbol	Description	Values, default
m	Number of clusters in the uniform-cluster or Zipf-cluster node distribution	20
σ	Standard deviation of Gaussian distribution	0.05
N	Number of nodes in the network	5k ~ 25k, 5k
M_2	Number of nodes in the first-stage sample set in Algorithm 2	50 ~ 250, 150
M_1	Number of nodes in the sample set in Algorithm 1	#of sample nodes in Algorithm 2
L	Number of closest local neighbors for computing local density	4
α	Zooming factor for the second-stage sampling in Algorithm 2	0.0001
w	Local window size for the second-stage sampling in Algorithm 2	0.005 ~ 0.1, 0.005

Table 6.1: Experimental parameters for estimating the network size.

In our experiments, we first run the density-based sampling algorithm, and then execute the uniform sampling algorithm with the same size of sample set. We also compare our two algorithms with the original one in [49, 69, 70], which uses only local density to estimate the global network size.

6.4.4 Experimental Results

We use the following metric as an estimation accuracy of the proposed approaches.

$$\text{Estimation accuracy ratio} = 1 - \frac{|N_{real} - N_{estimate}|}{N_{real}}, \quad (6.1)$$

where N_{real} is the real number of nodes in the network, and $N_{estimate}$ is our estimated number of nodes in the networks.

Figures 6.9, 6.10 and 6.11 compare three estimation schemes in the four node distribution. We see that the uniform sampling algorithm has the best estimation accuracy (almost above 0.9 in all node distribution patterns), while local density-based estimation approach only performs well with about 0.5 estimation accuracy when nodes are uniformly distributed. As nodes become more unevenly distributed, the density based approach performs worse. Overall, the density-based sampling algorithm performs worse than uniform sampling, but better than local density based approach. Since the size of the sample set is not large, not more than 5% of the total number of nodes in the network, the uniform sampling approach has achieved reasonably good estimation accuracy at reasonable cost. From 6.11, we find that both the uniform and density based sampling algorithms are not sensitive to the choices of L and w . Thus, we can choose the small L and w in the implementation of these two schemes.

We have not quantitatively analyzed for the sampling approaches so far. Here we give an intuitive explanation. Their theoretical analysis will be our future work. Since the total number of nodes is fixed, when some places contain more nodes, the number of nodes in other places will be reduced, while the average density remains fixed. The uniform sampling approach probes the space and computes the average density over the sample set, which is equivalent to probing a random point in a space with uniform node distribution. In contrast, the density based sampling approach may bias the calculated average density and lead to a poor performance.

6.5 Concluding Remarks

In this chapter, we have experimentally studied the routing issues in some small-world models, and investigated key techniques for applications of small-world models to the design of peer-to-peer networks. Our simulations have verified our theoretical results in Chapters 4 and 5. We have also improved Symphony's performance based on our proposed schemes in previous chapters. In addition, we have evaluated the routing issue in Watts-Strogatz's small-world model by experiments. Finally, we have proposed some sampling approaches for estimating the network size. Although the methods are simple, we believe that they are a basic tool for constructing peer-to-peer networks.

CHAPTER 6. EXPERIMENTAL STUDY OF SMALL-WORLD NETWORKS AND APPLICATIONS TO PEER-TO-PEER SYSTEMS

Figure 6.6: Average $D(p)/D(0)$, $C(p)/C(0)$ and $R(p)/R(0)$ for different n .

CHAPTER 6. EXPERIMENTAL STUDY OF SMALL-WORLD NETWORKS AND APPLICATIONS TO PEER-TO-PEER SYSTEMS

(a) $n = 200, k = 2$.

(b) $n = 200, k = 5$.

(c) $n = 200, k = 8$.

(d) $n = 200, k = 11$.

(e) $n = 200, k = 20$.

Figure 6.7: Average $D(p)/D(0)$, $C(p)/C(0)$ and $R(p)/R(0)$ for different k .

CHAPTER 6. EXPERIMENTAL STUDY OF SMALL-WORLD NETWORKS AND APPLICATIONS TO
PEER-TO-PEER SYSTEMS

(a) Uniform node distribution.

(b) Uniform-cluster node distribution (total number of clusters=20).

(c) Zipf-cluster node distribution (total number of clusters=20).

(d) One-cluster node distribution.

Figure 6.8: Examples of data generated for testing (total number of nodes=1000).

CHAPTER 6. EXPERIMENTAL STUDY OF SMALL-WORLD NETWORKS AND APPLICATIONS TO PEER-TO-PEER SYSTEMS

(a) Uniform node distribution.

(b) Uniform-cluster node distribution.

(c) Zipf-cluster node distribution.

(d) One-cluster node distribution.

Figure 6.9: Network size estimation with different choices of M_2 .

CHAPTER 6. EXPERIMENTAL STUDY OF SMALL-WORLD NETWORKS AND APPLICATIONS TO PEER-TO-PEER SYSTEMS

(a) Uniform node distribution.

(b) Uniform-cluster node distribution.

(c) Zipf-cluster node distribution.

(d) One-cluster node distribution.

Figure 6.10: Network size estimation for different choices of N .

CHAPTER 6. EXPERIMENTAL STUDY OF SMALL-WORLD NETWORKS AND APPLICATIONS TO PEER-TO-PEER SYSTEMS

(a) Uniform node distribution.

(b) Uniform-cluster node distribution.

(c) Zipf-cluster node distribution.

(d) One-cluster node distribution.

Figure 6.11: Network size estimation for different choices of w .

Chapter 7

ReCord: A Distributed Hash Table with Recursive Structure

In this chapter, we will propose a simple distributed hash table called ReCord, which is a generalized version of Randomized-Chord and offers improved tradeoffs in performance and topology maintenance over existing P2P systems. ReCord is scalable and can be easily implemented as an overlay network, and offers a good tradeoff between the node degree and query latency. For instance, an n -node ReCord with $O(\lg n)$ node degree has an expected latency of $\Theta(\lg n)$ hops. Alternatively, it can also offer $\Theta(\frac{\lg n}{\lg \lg n})$ hops latency at a higher cost of $O(\frac{\lg^2 n}{\lg \lg n})$ node degree. Meanwhile, simulations of the dynamic behaviors of ReCord are studied.

The remainder of this chapter is organized as follows. Section 7.1 presents the introduction. In Section 7.2, the construction of ReCord is described. Section 7.3 examines the bounds of the node degree and route path length. Section 7.4 presents an implementation of ReCord. Section 7.5 gives the simulation studies of ReCord's dynamic behaviors. Section 7.6 summarizes the finding and concludes this chapter.

7.1 Introduction

Peer to peer (P2P) networks have become popular in resource sharing applications recently. There have been millions of users in certain successful systems, such as Napster [79],

CHAPTER 7. RECORD: A DISTRIBUTED HASH TABLE WITH RECURSIVE STRUCTURE

Gnutella [42], and Kazaa [59]. P2P systems are distributed systems without any central authority or hierarchical organization, and each node in the systems performs similar functionality.

In order to efficiently locate an object in a large scale P2P system, many schemes rely on distributed hash tables (DHTs). Example systems include Chord [97], Pastry [91], CAN [87], and Tapestry [107]. A P2P system needs to consider the joining, departing of hosts, and the insertion/addition of resources, besides look up operation. DHT can be implemented as an overlay logical topology over the internet physical networks, where each node keeps the direct IP address of its neighbors in a routing table. Instead of connecting to all the other nodes in the system, each node in DHT only links to a small number of nodes. The key lies in ensuring a small *diameter* in the resulting overlay network. At the same time, DHT should allow new nodes to join or existing nodes to leave the system voluntarily, therefore, the cost of topology maintenance during this dynamic join-depart process should be kept as low as possible.

The following metrics are usually used to compare the performance and efficiency of the designed DHT.

- (a) **Degree and Diameter:** The number of links per node in a DHT should be small in order to reduce the cost of communication. Also, the diameter of the network should not be large in order to reduce the query latency.
- (b) **Scalability:** As the network size increases, the node degree, the query latency, and the traffic increased in queries should not increase drastically.
- (c) **Maintenance overhead:** When new nodes join or existing nodes depart, the overhead as measured in terms of the number of messages required to maintain the DHT should be as low as possible.

- (d) **Fault tolerance:** The DHT should be resilient to both node and link failures of the system. No matter what fraction of the nodes or links has failed, the data available in the remaining nodes should still be accessible.
- (e) **Load balance:** The resource keys should be evenly distributed over all nodes, and the traffic overhead resulted by query or maintenance operations should be balanced among nodes in the network.

In this chapter, we propose a simple distributed hash table, called ReCord, which is scalable and can be easily implemented as an overlay network. ReCord offers a good tradeoff between the node degree and query latency. For instance, an n -node ReCord with $O(\lg n)$ node degree has an expected latency of $\Theta(\lg n)$ hops. Alternatively, it can also offer $\Theta(\frac{\lg n}{\lg \lg n})$ hops latency at a higher cost of $O(\frac{\lg^2 n}{\lg \lg n})$ node degree.

7.2 Construction of ReCord

In this chapter, we will slightly abuse the notation of node identifiers and nodes themselves, and the same to resource key identifiers and resource themselves. Instead of mapping identifiers into m -bit numbers, we will map them into the unit circle ring $\mathcal{I} [0, 1)$, as with Symphony [70] and Viceroy [69]. By using a consistent hashing method, we can assume that both node and key identifiers are distributed evenly over the circle $[0, 1)$, and there is no collision.

Hashing the identifiers into ring $\mathcal{I} [0, 1)$ allows the identifier value to be independent of the maximum hashed space 2^m . Assume that the ring is formed in the clockwise direction. Denote the clockwise neighbor of node s on the ring by $SUCC(s)$, and denote its counter-clockwise neighbor by $PRED(s)$. A key x is stored at a nearest node y , where $y \geq x$ on the ring. We also call this node $SUCC(x)$.

The basic idea of ReCord is as follows. Suppose that there are totally n active nodes in a stable P2P system. Starting from any node s , divide the whole ring into k equal

Figure 7.1: An example of interval division ($k = 4$).

intervals, where $k > 1$ denotes an integer. Then divide the first interval closest to node s recursively until the length of the interval nearest to s is $\frac{1}{n}$, i.e. each interval at the deepest recursion level contains $O(1)$ nodes, and each interval at the second deepest recursion level contains $O(k)$ nodes, and each interval at the third deepest recursion level contains $O(k^2)$ nodes and so on, as shown in Figure 7.1.

The first division is also called level-1 division, and the next is called level-2 division, and so on. There are $c = \lg_k n$ such levels (assuming that $n = k^c$, where c denotes an integer). The length of each interval at level 1 is $\frac{1}{k}$, and $\frac{1}{k^2}$ for level 2, and $\frac{1}{k^i}$ at level i in general. The intervals at the same level are numbered sequentially clockwise along the ring. There are totally k intervals in every level. Based on the above definitions, for node s , we know that its interval j at level i corresponds to the range $[s + \frac{(j-1)k^i}{n}, s + \frac{jk^i}{n})$ on the ring. Randomly choose one node x in every interval, and set up a unidirectional connection from node s to x . We call the resulting network ‘ReCord’ for its recursive structure and similarity with Chord.

Comparing ReCord with Randomized-Chord, we find that in fact ReCord is a generalized version of Randomized-Chord. When $k = 2$, ReCord becomes Randomized-Chord.

7.3 Analysis

P2P systems have dynamic membership, meaning that a node may join and leave the network voluntarily. The number of the active nodes varies with the evolution of the P2P network. However, when we analyze the degree of each node or latency of each query, we suppose that the P2P network is static. Therefore, we will firstly analyze the tradeoffs

CHAPTER 7. ReCORD: A DISTRIBUTED HASH TABLE WITH RECURSIVE STRUCTURE

between link degree and query latency for ReCORD statically. Later, we will explain how to extend and implement it under the dynamic situation.

Theorem 7.1 *The node degree per node in an n -node ReCORD is $\Theta(k \lg_k n)$.*

Proof: Let $H(n)$ represent the number of links connected to an arbitrary node s in the n -node network. After the first division, there are $k - 1$ links, plus its links to nodes in the intervals included by level-2, hence we have the following relation:

$$H(n) = (k - 1) + H\left(\frac{n}{k}\right).$$

The solution of this recurrence is $H(n) = \Theta((k - 1) \lg_k n) = \Theta(k \lg_k n)$. Therefore, the degree of any node in ReCORD is bounded by $\Theta(k \lg_k n)$. ■

When $k = \Theta(1)$, the node degree in the n -node ReCORD is $H(n) = \Theta(\lg n)$. If $k = \Theta(\lg n)$, $H(n) = \Theta\left(\frac{\lg^2 n}{\lg \lg n}\right)$.

Before studying the query latency in ReCORD, we introduce the following lemma which will be used in the proof of the lower bound of the query latency.

Lemma 7.1 *Let X^m denote a random variable in the state space $0, 1, 2, \dots, m - 1$. Let*

$$Pr[X^m = i] = \begin{cases} \frac{k-1}{k^m-i}, & \text{when } 1 \leq i \leq m-1 \\ \frac{1}{k^m-1}, & \text{when } i = 0 \end{cases}$$

The expected time required for X^m to drop to 0 is lower bounded by $E[T_m] = \Omega(m)$

Proof: The proof is similar to that of Lemma 3.3 in [71], and hence omitted here. ■

Theorem 7.2 *Using the greedy routing algorithm, the expected path length per query in an n -node ReCORD is $\Theta(\lg_k n)$.*

Proof: Upper bound: Let $T(n)$ denote the number of hops required by a query. Consider the case when the message is routed to the 1st interval, according to the recursive

CHAPTER 7. RECORD: A DISTRIBUTED HASH TABLE WITH RECURSIVE STRUCTURE

construction, the time step required is $T(n) = T(\frac{n}{k})$. If the message is routed to interval j of the level-division ($1 < j \leq k$), in the worst case, the distance is reduced to $\frac{2n}{k} - 1$. In this case, after one more forwarding, the distance will be reduced to less than $\frac{n}{k}$, so the required time in the worst case is upper-bounded by $T(n) \leq 2 + T(\frac{n}{k})$. Since each link is connected to a randomly chosen node in each interval, the probability that the message is routed to interval 1 is $\frac{k}{n}$, and the probability that it is routed to intervals 2, 3, ..., k is $\frac{n-k}{n}$. Thus, an upper bound of the total expected number of steps is:

$$T(n) \leq \frac{1}{k}T(\frac{k}{n}) + \frac{k-1}{k}[2 + T(\frac{k}{n})]. \quad (7.1)$$

Solving Ineq. (7.1), we have $T(n) = O(\frac{2(k-1)}{k} \lg_k n) = O(\lg_k n)$. Therefore, for the greedy routing algorithm, the expected path length per query is $O(\lg_k n)$, where the constant multiplier in the formula is smaller than 2.

Lower bound: Suppose that all nodes in the ring are labelled by $0, 1, 2, \dots, n$. Node 0 is the destination, and node n is the source. We define *Phase* as follows: Phase 0 only contains node 0; Phase 1 consists of nodes in the interval $[1, k-1]$; Phase 2 consists of nodes in the interval $[k, k^2-1]$, and so on. Generally, Phase i contains nodes in the interval $[k^{i-1}, k^i-1]$. Suppose that there are in total m phases.

According to the division of intervals and randomly choosing one node among each interval, the probability that the message is routed to Phase $m-1$ is $\frac{k-1}{k}$, and $\frac{k-1}{k^2}$ if routed to Phase $m-2$, and so forth. Generally, the probability that the message is routed to Phase i is $\frac{k-1}{k^{m-i}}$, for $1 \leq i \leq m-1$, and $\frac{1}{k^{m-1}}$, for $i = 0$. By applying Lemma 7.1, we can deduce that the expected number of hops per query is $\Omega(m)$. There are totally $m = \lg_k n$ phases for n nodes. Therefore, the average number of hops per query is lower bounded by $\Omega(\lg_k n)$. ■

Our static analysis shows a good tradeoff between the node degree and the required hops per query. If we choose $k = \Theta(1)$, the node degree and query latency for an n -node

Figure 7.2: Impact of variable k on degree and latency.

ReCORD are $O(\lg n)$ and $\Theta(\lg n)$ respectively. If we let $k = \Theta(\lg n)$, the n -node ReCORD has $\Theta(\frac{\lg^2 n}{\lg \lg n})$ node degree and $\Theta(\frac{\lg n}{\lg \lg n})$ query latency. Figure 7.2 shows the trade-off between the node degree and query latency, given the total number of active nodes is $n = 2^{15} = 32768$. Figure 7.2 shows that the node degree increases almost linearly as k increases, but the query latency drops quickly within a small range of k .

7.4 Implementation of ReCORD

7.4.1 Estimation of network size

Although the analytical results in the previous section can be directly applied to a static network, the derived bounds for degree and latency are not as tight as the case using active nodes in the real P2P network. Now we suppose a dynamic environment, where the nodes join and leave dynamically. The main difficulty of this extension is that for each node, it requires a rather accurate information of the global network size for the construction of links. When we divide network in each level, we need to know the value of n , the total number of active nodes.

CHAPTER 7. RECORD: A DISTRIBUTED HASH TABLE WITH RECURSIVE STRUCTURE

Currently, most estimation processes uses the density information around the vicinity of the estimating node [69, 70, 49]. Let L_f denote the fraction length of an interval including f distinct nodes. The network size can be estimated by $\frac{f-1}{L_f}$. In [69, 49], the length between estimating node and its successor is used to estimate the size of the overall network. Symphony [70] applies the length between estimating node's predecessor and successor in estimation procedure, and its experiment shows that the impact of increasing f on average latency is not significant.

Other methods can be also applied to estimate network size, such as through a central server, or piggybacking along lookup messages [70], or randomly choosing several pairs of continuous nodes, and using their average length for estimation.

Knowing the network size is an important step for dynamic network construction. In our experiments, as with Symphony, we use $f = 3$ to estimate the global network size in ReCord.

7.4.2 Join and Leave maintenance

7.4.2.1 Join Protocol

Suppose that a new node s joins the network through an existing node. Firstly, node s chooses its identifier from $[0, 1)$ uniformly at random. Secondly, node s is inserted between $PRED(s)$ and $SUCC(s)$, and runs the estimation protocol, and updates the estimated network size $\tilde{n}_s$ for all three nodes. Next, it divides the whole ring $[0, 1)$ recursively into intervals $[s + \frac{(j-1)k^i}{\tilde{n}_s}, s + \frac{jk^i}{\tilde{n}_s})$ starting from s as described in Section 7.2. Then it sets up one link to a node randomly chosen from each interval. The implementation detail for link construction is that it first generates a random real number x in the interval $[s + \frac{(j-1)k^i}{\tilde{n}_s}, s + \frac{jk^i}{\tilde{n}_s})$, then looks up $SUCC(x)$. If $SUCC(x)$ is in the range $[s + \frac{(j-1)k^i}{\tilde{n}_s}, s + \frac{jk^i}{\tilde{n}_s})$, the connection is built successfully, otherwise, it has to re-establish the link for the interval. In order to avoid flooding traffic made by link reestablishment, we limit the times of reestablishment. If node s still can't find a node in an interval after q times tries, we let

it give up the link construction for this interval. The value of q should be related to the interval length. More details will be shown in the experiment part.

Similar to Symphony, ReCord also bounds the number of incoming links per node, which is good for load balancing of the whole network. Once the number of incoming links of a node has reached $2 \lg_k n$, any new request to establish a link with it will be rejected. The requesting node has to make another attempt.

Since node s needs a lookup operation that requires $O(\lg_k n)$ messages for each link establishment, the whole cost of $O(k \lg_k n)$ link constructions is $O(k \lg_k^2 n)$ messages.

7.4.2.2 Leave Protocol

Once node s leaves the system, all its outgoing links will be snapped. Its predecessor and successor nodes need to reinstate their links, and corresponding neighbor nodes need to update their estimation of network size. At the same time, all the incoming links of node s are broken, and corresponding connection nodes need to re-select another node randomly in the same interval as node s is located in. This operation can be triggered by the periodic detections by nodes connected to node s .

If node s leaves voluntarily, it will gracefully inform related nodes to update their connection information, otherwise, the connection information has to be updated when the other nodes have periodically detected the failure of node s . Other details of this protocol are similar to that in Chord.

7.4.3 Re-linking operation

The total number of active nodes in the P2P network always changes as the network expands or shrinks. When node s finds that its current estimated network size $\tilde{n}_s$ is not equal to its stored stale estimation $\tilde{n}'_s$, it needs to re-build its links to nodes in the new intervals. One conservative solution is to re-link every construction on every update of $\tilde{n}_s$. In this way, the traffic resulted by the re-linking operation would be excessive. According

to analyzed bounds in Section 7.3, the messages required for one node to re-link all its connections are $O(k \log_k^2 n)$.

In order to avoid the excessive traffic resulted from re-linking operation, and guarantee the stability of the whole P2P network, we apply the same re-linking criterion as in Symphony: re-linking operation occurs only when $\frac{\tilde{n}_s}{n'_s} \geq 2$ or $\frac{\tilde{n}_s}{n'_s} \leq \frac{1}{2}$, where $\tilde{n}_s$ is node's updated estimated network size, and $\tilde{n}'_s$ is node's stored stale network size.

7.4.4 Fault tolerance

In Chord (or Koorde with constant-degree), each node keeps a list of successors to increase the system's robustness: each node maintains r connections of its immediate succeeding nodes rather than only one immediate successor. Certainly, it will keep the whole P2P network more robust, but it also requires some extra operations and corresponding additional cost to maintain the successor list. Using a similar scheme, Symphony makes r copies of a node's content at each of r succeeding nodes. Other DHTs, such as CAN, Pastry, and Tapestry keep several backup links for each node.

Compared with the above DHTs, we found that ReCord has a natural structure for fault tolerance: at the last dividing level, each node is already connected to its k following succeeding nodes, which is equal to a successor list in Chord. ReCord need not keep any backup link or redundant links to increase the robustness of the whole system. Therefore, it entails no extra overhead to offer fault tolerance.

As stated in [57], in order to keep live nodes connected in cases of nodes failures, some nodes need to have node degree of at least $\Omega(\lg n)$. Moreover, the experiments in [70] show that the local links are crucial for maintaining connectivity of P2P topology. By construction, ReCord has rich "local" connections and relatively sparse "long" connections. Our experimental results, to be presented shortly, confirm that this is indeed a good strategy for forming the connectivity of the P2P network.

7.5 Experiments

Based on the implementation described in the preceding section, we run the simulation of ReCORD with nodes ranging from 2^4 to 2^{15} . The impacts of different parameters shown in the simulation results are analyzed.

We focus on four types of network. The first one is a *static* network, in which the global network size is known for every node. The second one is called an *expanding* network, where the rate of node joining is higher than that of node departure in a given unit time interval. The third one, called a *shrinking* network, is opposite of an expanding network. The last one is called a *stable* network, in which the rate of node joining is equal to that of node departure nodes in a unit time interval.

7.5.1 Estimation of network size

For each node, we let it use the density between its predecessor and successor to estimate the global network size. Figure 7.3 shows the comparisons between the estimated and the actual network size for both small and large P2P system which is in the steady state. For the small scale network with $n = 250$ active nodes, the difference between estimated $\lg \tilde{n}$ and actual $\lg n$ is no more than 4. For a larger scale network with $n = 11,374$ active nodes, the difference between estimated $\lg \tilde{n}$ and actual $\lg n$ is no more than 8. In either network, the difference between the estimated and actual $\lg n$ for most nodes is smaller than 4. This shows that the approximation is accurate enough for the construction of ReCORD.

Figure 7.4 shows the average estimation of network size over all nodes of the expanding, shrinking, and stable networks respectively over time. The comparisons of average $\lg \tilde{n}$ and actual $\lg n$ are shown.

CHAPTER 7. REcORD: A DISTRIBUTED HASH TABLE WITH RECURSIVE STRUCTURE

Figure 7.3: Estimation of network size for stable P2P systems.

Figure 7.4: Estimation of network size for expanding, shrinking, and stable network respectively.

CHAPTER 7. ReCORD: A DISTRIBUTED HASH TABLE WITH RECURSIVE STRUCTURE

Figure 7.5: Average degree and latency on expanding network.

7.5.2 Degree and Latency

Figure 7.5 shows the average node degree and latency over an expanding network with different k values.

Figure 7.6 shows the tradeoff between degree and latency over different k values in a stable P2P network, given that the number of active nodes is in the range of $[2070, 2121]$. For Figure 7.6, $k = 5$ is obviously the best choice for the P2P network of those sizes. We can also see that it fits the analysis (cf. Figure 7.2) quite well.

7.5.3 Fault tolerance

Figure 7.7 shows how the fraction of failure links will influence the query latency. Three cases: $k = 3$, $k = 5$, and $k = 7$ were run under the stable environment. According to Figure 7.2, the node degree increases as k increases for a fixed n . However, from Figure 7.7,

Figure 7.6: Tradeoff between degree and latency over different choices of k in a stable network (the number of active nodes is between [2070,2121]).

we can see that independent of k , only when more than half of the links fail, the query latency is adversely affected.

7.6 Concluding Remarks

We have proposed a simple DHT topology, called ReCord, which is a generalization of Randomized-Chord [71], in the sense that Randomized-Chord is a special case of ReCord when $k = 2$. ReCord offers a good tradeoff between the node degree and query latency: an n -node ReCord with $O(\lg n)$ node degree has an expected latency of $\Theta(\lg n)$ hops. Alternatively, it can also offer $\Theta(\frac{\lg n}{\lg \lg n})$ hops latency at a higher cost of $\Theta(\frac{\lg^2 n}{\lg \lg n})$ node degree. Some implementation techniques of ReCord are presented, including the estimation of network size, join and departure maintenance, re-linking operations, etc. Meanwhile, simulations of the dynamic behaviors of ReCord are studied.

In actual P2P systems, the different bounds of degree and latency of the constructed networks usually offer different insights of DHTs. Lower degree decreases the number of open connections and the protocol traffic made by the pinging operation; the number of neighbors whose states need to be changed when a node joins or leaves is also smaller.

CHAPTER 7. RECORD: A DISTRIBUTED HASH TABLE WITH RECURSIVE STRUCTURE

Figure 7.7: Impact of link failures on the average latency.

However, the lower connectivity of the low node degree also means that the network is easy to split up, and hence it has weak fault tolerance. On the other hand, higher node degree leads to better connectivity and reduces the network diameter, the longest path length for the query operation. The lower query latency also leads to lower joining and departure costs. Since the join and leave operation will make use of the query operation, the small path latency will also decrease the cost of joining and leaving. We can adjust the k value to suit the P2P networks required for different environments. A more proactive and perhaps more useful method is to dynamically monitor the P2P system, e.g. sensing the frequency of the nodes joining and leaving, and adjusting the k value dynamically in real time. We will extend our work in this respect in our future research.

Chapter 8

Discussions and Conclusion

8.1 Summary of Contributions

In this thesis, we have proposed a constructive model for scale-free networks that use partial information, extended small-world models with near-optimal and optimal routing algorithms, and a distributed hash table with recursive structure. Our main contributions are listed as follows.

- (1) *A construction model of scale-free networks*¹. The novelty of our model lies in that each node establishes the connections only based on the information of a small sample set of the entire network. We have demonstrated the degree distribution of our model both theoretically and experimentally. We have rigorously proved that the logarithmic size of the sample set can guarantee the emergence of the power law of degree distribution. Our construction model can better approximate the evolution of the real-world scale-free networks than the existing models. Our results can also be applied to the constructions of peer-to-peer networks, where the global information is generally unavailable for any individual node in the network.
- (2) *Near-optimal and optimal routing for small-world networks with augmented local awareness*². We have provided a new perspective for improving the decentralized

¹The results are published in [105]

²The results are published in [101, 104, 102]

routing in Kleinberg’s small-world networks: adding the augmented local links and increasing the augmented local awareness. We have shown that by using this method, both near-optimal algorithm with $O(\lg n \lg \lg n)$ expected routing complexity and $O(\lg^2 n)$ bits of space requirement per node, and optimal algorithm with $O(\lg n)$ expected routing complexity and poly-logarithmic bits of space requirement per node can be achieved. Our near-optimal algorithm outperforms the existing schemes for Kleinberg’s small-world networks, given the same amount of information stored on each node. Our optimal algorithm, to the best of our knowledge, is the first to achieve the optimal routing complexity while still keeping a poly-logarithmic number of bits of information on each node in the small-world networks. Our findings have implied that the augmented local connections indeed can make small-world networks more navigable. We have also applied our results to improve the performance of Symphony [70].

- (3) *A distributed hash table with recursive structure for peer-to-peer networks*³. We have developed a generalized version of Randomized-Chord, called ReCord, which is a simple and scalable distributed hash table. We have demonstrated that ReCord offers good tradeoffs between performance and topology maintenance: providing an expected latency of $\Theta(\lg n)$ hops with node degree $O(\lg n)$, or $\Theta(\frac{\lg n}{\lg \lg n})$ hops latency with a higher cost of $O(\frac{\lg^2 n}{\lg \lg n})$ node degree. ReCord provides an adaptable infrastructure which can adjust the real-time network connections by monitoring the peer-to-peer system dynamically.

8.2 Future Work

Although many models or structures have been proposed for current large-scale distributed systems, much work remains to be done. We list the following open problems for future research:

³The results are published in [103]

CHAPTER 8. DISCUSSIONS AND CONCLUSION

- (1) *Construction of small-world networks without global information of network size.* The existing navigable small-world networks, e.g., Kleinberg's small-world network and its extensions, assume that each node knows about the size of the entire network. In practice, it is an unrealistical assumption and hence is not suitable for the construction of large-scale distributed systems. It will be of great significance to develop a small-world network that offers both efficient decentralized routing and yet require no information about the network size.

Although Symphony [70] has successfully applied Kleinberg's small-world model to the design of peer-to-peer overlay networks, it depends on the preprocessing that nodes are uniformly hashed into a ring space to precisely estimate the global network size. Unfortunately, the uniform hashing can not preserve the proximity order of nodes, which makes it difficult for Symphony to support complex and partial match queries.

- (2) *Heterogeneous peer-to-peer systems.* Most existing peer-to-peer systems assume that all nodes play the same roles. In real world, however, the voluntariness, capacity, service requirement, bandwidth, transmission rate for each node may differ considerably from another. Any protocol that regards each node as equivalent to another will lead to wastage of resources and load imbalance of the networks. Thus, a promising research topic is to develop a heterogeneous peer-to-peer infrastructure that best exploits the network resources.

A natural solution may be assigning each node an initial attractiveness value once it joins the network, and allowing the probability that it gets the connections from the other nodes to be proportional to its attractiveness value. This may lead to a power law random graph, and we may then apply the current results on scale-free networks to the system implementation and analysis. A difficulty with current evolution models of scale-free networks is that they only consider the process of node

insertions. Thus, a useful research direction is to develop random graph evolution models that can capture the full dynamic features of peer-to-peer systems, and can support both node joining and departure operations.

(3) *Mapping between the peer-to-peer logical overlay and the underlying physical network.*

In most existing peer-to-peer structures, e.g., Chord [97], CAN [87], etc., the virtual overlay topology does not match with the underlying physical topology, and the mismatch incurs extra routing costs. An efficient method for reducing the mismatch is to explore the awareness of physical topology for constructing the logical virtual overlay. Although the schemes in [88, 81] can be used to help explore the topology awareness, they are not scalable as the network size increases. This topic requires further research.

(5) *Applications of game theory in peer-to-peer systems.*

In distributed systems without centralized authority, such as peer-to-peer networks, each peer is regarded as a rational and strategic player, and acts in a distributed and selfish fashion so that each peer can maximize its own utility. From the game theory point of view, when peers can cooperate with others at a certain degree, the system may finally meet an equilibrium point in which no peer can improve its utility by deviating from the current strategy. From the viewpoint of the whole system, the design goal is to maximize the system utility. Therefore a promising research direction is to explore game theory and develop distributed algorithms in which each peer can maximize its own utility, while realizing optimal welfare of the system.

8.3 Concluding Remarks

The results in this thesis have provided novel perspectives for the constructions of large-scale distributed networks. I am confident that the new insights will help better understand

CHAPTER 8. DISCUSSIONS AND CONCLUSION

the evolutions of real-world distributed networks and contribute to their further applications.

Appendix A Pseudocodes for ReCord Implementation

Algorithm 1 Estimation of network size

EstimationNetworkSize(s) // estimate the global network size for node s

```

1:  $length \leftarrow$  the clockwise length between  $SUCC(s)$  and  $PRED(s)$ ; // length of the whole ring is 1
2:  $estimated\_number \leftarrow \frac{2}{length}$ ;
3: return;  $estimated\_number$ 

```

Algorithm 2 Establishment of links for each node

BuildLinks($s, k, length$) /*build all links for node s . k is the number phases per level. $length$ is the length of division level. Initially call BuildLinks($s, k, 1$)*/

```

1: if  $(length/k) < (1/EstimationNetworkSize(s))$  then
2:   return;
3: end if
4: for  $(j = 2; j \leq k - 1; j++)$  do
5:   for  $(retry\_time = 1; retry\_time \leq \sqrt{EstimationNetworkSize(s) * length/k}; retry\_time++)$  do {
      $\sqrt{EstimationNetworkSize(s) * length/k}$  is the bound of retry times }
6:      $rand \leftarrow$  a random number between  $[0, 1)$ ;
7:      $temp\_n \leftarrow FindSuccessor(temp\_n, s)$ ;
8:     if incoming degree of node  $temp\_n \leq 2 * k * \log(EstimationNetworkSize(s))$  then {  $2 * k * \log(EstimationNetworkSize(s))$  is the limitation of incoming degree per node }
9:       if  $rand \in [ID(s) + \frac{j * (length/k)}{EstimationNetworkSize(s)}, ID(s) + \frac{(j+1) * (length/k)}{EstimationNetworkSize(s)})$  then {  $ID(s)$  the identifier of node after mapped into  $[0, 1)$  }
10:        build a connection between node  $s$  and node  $temp\_n$ ;
11:      end if
12:    end if
13:  end for
14: end for
15: BuildLinks( $s, k, length/k$ )

```

Algorithm 3 Find the successor for a given identifier

FindSuccessor(s, x) /*find the successor of identifier x through node s */

```

1: if  $x \in [ID(s), ID(s+1))$  then {  $ID(s)$  is the identifier of node  $s$  mapped into  $[0, 1)$  }
2:   return  $s + 1$ ;
3: else
4:   return FindSuccessor( $s + 1, x$ )
5: end if

```

APPENDIX A. PSEUDOCODES FOR RECORD IMPLEMENTATION

Algorithm 4 Find the predecessor for a given identifier

```

FindPredecessor(s,x) /*find the predecessor of identifier x through node s*/
1: if  $x \in [ID(s), ID(s-1))$  then { $ID(s)$  is the identifier of node  $s$  mapped into  $[0, 1)$ }
2:   return  $s-1$ ;
3: else
4:   return FindSuccessor( $s-1, x$ )
5: end if

```

Algorithm 5 Join operation

```

Join( $n_0, x$ ) /*node  $s$  joins the system through node  $n_0$ */
1:  $n_1 \leftarrow FindPredecessor(n_0, s)$ ;
2:  $SUCC(n_1) \leftarrow s$ ;
3:  $PRED(s) \leftarrow n_1$ ;
4:  $n_2 \leftarrow FindSuccessor(n_0, s)$ ;
5:  $SUCC(s) \leftarrow n_2$ ;
6:  $PRED(n_2) \leftarrow s$ ;
7: EstimationNetworkSize( $s$ )
8: BuildLinks( $s, k, 1$ )

```

Algorithm 6 Leave operation

```

Leave( $x$ )
1:  $SUCC(PRED(s)) \leftarrow SUCC(s)$ ;
2:  $PRED(SUCC(s)) \leftarrow PRED(s)$ ;
3: delete all incoming and outgoing links of node  $s$  and inform corresponding nodes;

```

Algorithm 7 Lookup operation

```

Lookup( $x$ ) /* lookup identifier  $x$  through node  $s$  */
1:  $min\_length \leftarrow 0$ ;
2:  $min\_neighbor \leftarrow SUCC(s)$ ;
3: if FindSuccessor( $s, x$ )= $s$  then
4:   return 0;
5: end if
6:  $min\_length \leftarrow$  clockwise distance between  $min\_neighbor$  and FindSuccessor( $x$ );
7: for  $n_i \in$  neighbors of nodes  $s$  do
8:   if clockwise distance between  $n_i$  and FindSuccessor( $x$ ); $min\_length$  then
9:      $min\_length \leftarrow$  clockwise distance between  $n_i$  and FindSuccessor( $x$ );
10:     $min\_neighbor \leftarrow n_i$ ;
11:   end if
12: end for
13: return Lookup( $min\_neighbor, x$ )+1;

```

Algorithm 8 Relinking operation

```

Relink( $s$ )
1: if  $\frac{\tilde{n}}{n'} < 0.5$  or  $\frac{\tilde{n}}{n'} > 2$  then { $\tilde{n}$  denotes new estimated  $n$ ;  $n'$  denotes old estimated  $n$ }
2:   delete all outgoing links of node  $s$  and inform corresponding nodes;
3:   BuildLinks( $s, k, 1$ )
4: end if

```

References

- [1] L. Adamic, R. Lukose, and A. Puniyani. Search in Power-Law Networks. *Physical Review E*, 64:046135, 2001.
- [2] W. Aiello, F.R.K. Chung, and L. Lu. A Random Graph Model for Massive Graphs. In *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing*, pages 171–180, 2000.
- [3] R. Albert and A.-L. Barabási. Statistical Mechanics of Complex Networks. *Reviews of Modern Physics*, 74:47–97, 2002.
- [4] R. Albert, H. Jeong, and A.-L. Barabási. The Diameter of the World Wide Web. *Nature*, 401(9):130–131, 1999.
- [5] R. Albert, H. Jeong, and A.-L. Barabási. Error and Attack Tolerance in Complex Networks. *Nature*, 406:378–382, 2000.
- [6] N. Alon and J.H. Spencer. *The Probabilistic Method*. John Wiley, 2000.
- [7] S. Androutsellis-Theotokis and D. Spinellis. A Survey of Peer-to-Peer Content Distribution Technologies. *ACM Computing Surveys*, 36(4):335–371, 2004.
- [8] J. Aspnes, Z. Diamadi, and G. Shah. Fault-tolerant Routing in Peer-to-Peer Systems. In *Proceedings of the 21st ACM Symposium on Principles of Distributed Computing*, pages 223–232, 2002.
- [9] B. Bollobás and F.R.K. Chung. The Diameter of a Cycle Plus a Random Matching. *SIAM Journal on Discrete Mathematics*, 1(3):328–333, 1988.
- [10] A.-L. Barabási and R. Albert. Emergence of Scaling in Random Networks. *Science*, 286:509–512, 1999.

REFERENCES

- [11] L. Barrière, P. Fraigniaud, E. Kranakis, and D. Krizanc. Efficient Routing in Networks with Long Range Contacts. In *Proceedings of the 15th International Symposium on Distributed Computing*, pages 270–284, 2001.
- [12] N. Berger, B. Bollobás, C. Borgs, J. Chayes, and O. Riordan. Degree Distribution of the FKP Network Model. In *Proceedings of the 13th International Colloquium on Automata, Languages and Programming*, pages 725–738, 2003.
- [13] N. Berger, C. Borgs, J.T. Chayes, R.M. D’Souza, and R.D. Kleinberg. Degree Distribution of Competition-Induced Preferential Attachment Graphs. *Combinatorics, Probability and Computing*, 2005. To appear.
- [14] B. Bollobás and O. Riordan. *Mathematical Results on Scale-Free Random Graphs*. In *Handbook of Graphs and Networks*, 2002.
- [15] B. Bollobás and O. Riordan. Coupling Scale-Free and Classical Random Graphs. *Internet Mathematics*, 1(2):215–225, 2003.
- [16] B. Bollobás and O. Riordan. Robustness and Vulnerability of Scale-Free Random Graphs. *Internet Mathematics*, 1(2):1–35, 2003.
- [17] B. Bollobás and O. Riordan. The Diameter of a Scale-Free Random Graph. *Combinatorica*, to appear.
- [18] B. Bollobás, O. Riordan, J. Spencer, and G. Tusnády. The Degree of Sequence of a Scale-Free Random Graph Process. *Random Structures and Algorithms*, 18:279–290, 2001.
- [19] L. Bresau, P. Cao, L. Fan, G. Phillips, and S. Shenker. Web Caching and Zipf-Like Distributions: Evidence and Implications. In *Proceedings of the 18th Annual Joint Conference of the IEEE Computer and Communications Societies*, pages 126–134, 1999.
- [20] A. Broder, R. Kumar, F. Maghoul, P. Raghavan S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener. Graph Structure in the Web. In *Proceedings of the 9th International World Wide Web Conference*, pages 309–320, 2002.
- [21] G. Buckley and D. Osthus. Popularity Based Random Graph Models Leading to a Scale-Free Degree Distribution, 2001. Submitted.

REFERENCES

- [22] I. Clarke, O. Sandberg, B. Wiley, and T. W. Hong. Freenet: A Distributed Anonymous Information Storage and Retrieval System. In *Proceedings of the International Workshop on Design Issues in Anonymity and Unobservability*, pages 46+, 2001.
- [23] E. Cohen, H. Kaplan, and A. Fiat. Associative Search in Peer-to-Peer Networks: Harnessing Latent Semantics. In *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies*, pages 1261–1271, 2003.
- [24] Grid computing. <http://www.gridbus.org/>.
- [25] C. Cooper and A. Frieze. A General Model of Web Graphs. *Random Structures and Algorithms*, 22:311–335, 2003.
- [26] D. Coppersmith, D. Gamarnik, and M. Sviridenko. The Diameter of a Long-Range Percolation Graph. *Random Structures and Algorithms*, 21(1):1–13, 2002.
- [27] D. Watts and S. Strogatz. Collective Dynamics of Small-World Networks. *Nature*, 393:440–442, 1998.
- [28] P. Dodds, R. Muhamad, and D. Watts. An Experimental Study of Search in Global Social Networks. *Science*, 301:827–829, 2003.
- [29] S.N. Dorogovtsev and J.F.F. Mendes. Evolution of Networks. *Advances in Physics*, 51:1079–1187, 2002.
- [30] S.N. Dorogovtsev, J.F.F. Mendes, and A.N. Samukhin. Structure of Growing Networks with Preferential Linking. *Physical Review Letters*, 85(21):4633–4636, 2000.
- [31] E. Drinea and M. Enachescu and M. Mitzenmacher. Variations on Random Graph Models for the Web. Technical report, Department of Computer Science, Harvard University, 2001.
- [32] M. Kochen Ed. *The Small World*. Ablex, Norwood, 1989.
- [33] S. El-Ansary, L.O. Alima, P. Brand, and S. Haridi. A Framework for Peer-To-Peer Lookup Services Based on k -ary Search. Research Report T2002-06, Department of Microelectronics and Information Technology, Royal Institute of Technology, Stockholm, Sweden, 2002.

REFERENCES

- [34] A. Fabrikant, E. Koutsoupias, and C. Papadimitriou. Heuristically Optimized Trade-Offs: a New Paradigm for Power Laws in the Internet. In *Proceedings of the 29th International Colloquium on Automata, Languages and Programming*, pages 110–122, 2002.
- [35] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On Power-Law Relationships of the Internet Topology. In *Proceedings of ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pages 251–262, 1999.
- [36] A.D. Flaxman, A.M. Frieze, and J. Vera. A Geometric Preferential Attachment Model of Networks. In *Proceedings of the 3rd International Workshop on Algorithms and Models for the Web-Graph*, pages 44–55, 2004.
- [37] I. Foster. Internet Computing and the Emerging Grid. *Nature Web Matters*, December, 2000. Available at <http://www.nature.com/nature/webmatters/grid/grid.html>.
- [38] I. Foster, C. Kesselman, and S. Tuecke. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International Journal of Supercomputer Applications*, 15(3):200–222, 2001.
- [39] P. Fraigniaud, C. Gavoille, and C. Paul. Eclecticism Shrinks Even Small Worlds. In *Proceedings of the 23rd Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing*, pages 169–178, 2004.
- [40] P. Ganesan and G.S. Manku. Optimal Routing in Chord. In *Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 176–185, 2004.
- [41] Genome@home. The Genome@home project web site. <http://genomeathome.stanford.edu/>. 2003.
- [42] Gnutella. <http://gnutella.wego.com>.
- [43] J. Gómez-Gardeñes and Y. Moreno. Local versus Global Knowledge in the Barabási-Albert Scale-Free Network Model, Preprint, arXiv:cond-mat/0401269. Available at <http://www.arxiv.org>.
- [44] DSS Group. Gnutella: To the Bandwidth Barrier and Beyond. <http://dss.clip2.com>, Nov. 6, 2000.

REFERENCES

- [45] K. Gummadi, R. Gummadi, S. Gribble, S. Ratnasamy, S. Shenker, and I. Stoica. The Impact of DHT Routing Geometry on Resilience and Proximity. In *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, pages 381–394, 2003.
- [46] H. Jeong and B. Tombor and R. Albert and Z.N. Oltvai and A.-L. Barabási. The Large-Scale Organization of Metabolic Networks. *Information Processing Letters*, 33:305–308, 1990.
- [47] A. Halevy, Z. Ives, P. Mork, and I. Tatarinov. Piazza: Data Management Infrastructure for Semantic Web Applications. In *Proceedings of the 12th International World Wide Web Conference*, pages 556–567, 2003.
- [48] C.M. Homan and G. Istrate. Small worlds, locality, and flooding on landscapes. Research Report TR-2003-796, Department of Computer Science, University of Rochester, USA, 2003.
- [49] K. Horowitz and D. Malkhi. Estimating Network Size from Local Information. *Information Processing Letters*, 88(5):237–243, 2003.
- [50] R. Huebsch, J.M. Hellerstein, N.L. Boon, T. Loo, S. Shenker, and I. Stoica. Querying the Internet with PIER. In *Proceedings of the 19th International Conference on Very Large Databases*, pages 321–332, 2003.
- [51] I. Benjamini and N. Berger. The Diameter of Long-Range Percolation Clusters on Finite Cycles. *Random Structures and Algorithms*, 19(2):102–111, 2001.
- [52] A. Iamnitchi, M. Ripeanu, and I. Foster. Locating Data in (Small-Wrold?) Peer-to-Peer Seicentic Collaborations. In *Proceedings of the second International Workshop on Peer-to-Peer Systems*, pages 232–241, 2002.
- [53] A. Iamnitchi, M. Ripeanu, and I. Foster. Small-Wrold File-Sharing Communities. In *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies*, pages 952–963, 2004.
- [54] J. Balthrop and S. Forrest and M.E.J. Newman and M.M. Williamso. Technological Networks and the Spread of Computer Viruses. *Science*, 304:527–529, 2004.
- [55] Jabber. The Jabber web site. <http://www.jabber.org/>. 2003.

REFERENCES

- [56] M.A. Jovanovic, F.S. Annexstein, and K.A. Berman. Scalability Issues in Large Peer-to-Peer Networks - A Case Study of Gnutella. Technical report, University of Cincinnati, 2001.
- [57] M. F. Kaashoek and David R. Karger. Koorde: A Simple Degree-optimal Distributed Hash Table. In *Proceedings of the second International Workshop on Peer-to-Peer Systems*, pages 98–107, 2003.
- [58] R. Kasturirangan. Multiple Scales in Small-World Graphs. *Preprint, arXiv cond-mat/9904055. Available at <http://www.arxiv.org>.*
- [59] Kazaa. <http://www.kazaa.com>.
- [60] D. Kemper, J. Kleinberg, and A. Demers. Spatial Gossip and Resource Location Protocols. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, pages 163–172, 2001.
- [61] J. Kleinberg. The Small-World Phenomenon: An Algorithmic Perspective. In *Proceedings of the 32nd ACM Symposium on Theory of Computing*, pages 163–170, 2000.
- [62] J. Kleinberg. Small-World Phenomena and the Dynamics of Information. In *Proceedings of the 15th Neural Information Processing Systems*, 2001. Available at <http://www.nips.cc/NIPS2001/papers/>.
- [63] J. Kubiawicz, D. Bindel, Y. Chen, P. Eaton, D. Geels, R. Gummadi, S. Rhea, H. Weatherspoon, W. Weimer, C. Wells, and B. Zhao. Oceanstore: An architecture for global-scale persistent storage. In *Proceedings of the 9th International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 190–201, 2000.
- [64] R. Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal. Stochastic Models for the Web Graph. In *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, pages 57–65, 2000.
- [65] E. Lebar and N. Schabanel. Almost Optimal Decentralized Routing in Long-Range Contact Networks. In *Proceedings of the 31st International Colloquium on Automata, Languages and Programming*, pages 179–188, 2004.

REFERENCES

- [66] N. Leibowitz, M. Ripeanu, and A. Wierzbicki. Deconstructing the Kazaa Network. In *Proceedings of the 3rd IEEE Workshop on Internet Applications*, pages 112–120, 2003.
- [67] M. Li, W.-C. Lee, and A. Sivasubramaniam. Semantic Small World: An Overlay Network for Peer-to-Peer Search. In *Proceedings of the 12th IEEE International Conference on Network Protocols*, pages 228–238, 2004.
- [68] M. Biskup. Graph Diameter in Long-Range Percolation. *Submitted to Electronic Journal of Probability*, 2004.
- [69] D. Malkhi, M. Naor, and D. Ratajczak. Viceroy: A Scalable and Dynamic Emulation of the Butterfly. In *Proceedings of the 21st ACM Symposium on Principles of Distributed Computing*, pages 183–192, 2002.
- [70] G. S. Manku, M. Bawa, and P. Raghavan. Symphony: Distributed Hashing In A Small World. In *Proceedings of the 4th USENIX Symposium on Internet Technologies and Systems*, pages 127–140, 2003.
- [71] G.S. Manku, M. Naor, and U. Wieder. Know thy Neighbor’s Neighbor: The Power of Lookahead in Randomized P2P Networks. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing 2004*, pages 54–63, 2004.
- [72] C. Martel and V. Nguyen. Analyzing Kleinberg’s (and other) Small-World Models. In *Proceedings of the 23rd Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing*, pages 179–188, 2004.
- [73] C. McDiarmid. *Concentration*. In M. Habib, C. McDiarmid, J. Ramirez-Alfonsin, and B. Reed (eds.), *Probabilistic Methods in Algorithmic Discrete Mathematics*, pages 195–248, Springer, 1998.
- [74] M.E.J. Newman. Models of the Small World. *Journal of Statistical Physics*, 101:819–841, 2000.
- [75] M.E.J. Newman and D.J. Watts. Scaling and Percolation in the Small-World Network Model. *Physical Review E*, 60:7332–7342, 1999.
- [76] M.A. De Menezes, C.F. Moukarzel, and T.J.P. Penna. First-order Transition in Small-World Networks. *Preprint, arXiv cond-mat/9903426*. Available at <http://www.arxiv.org>.

REFERENCES

- [77] M. Mihail, C. Papadimitriou, and A. Saberi. On Certain Connectivity Properties of the Internet Topology. In *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*, pages 28–35, 2003.
- [78] S. Mossa, M. Barthélemy, H.E. Stanley, and L.A.N. Amaral. Truncation of Power Law Behavior in “Scale-Free” Network Models Due to Information Filtering. *Physical Review Letters*, 88(13):138701, 2002.
- [79] Napster. <http://www.napster.com>.
- [80] M.E.J. Newman. The Structure and Function of Complex Networks. *Preprint, arXiv cond-mat/0303516*. Available at <http://www.arxiv.org>.
- [81] T. Ng and H. Zhang. Towards Global Network Positioning. In *Proceedings of the 1st ACM SIGCOMM Internet Measurement Workshop*, pages 25–29, 2001.
- [82] V. Nguyen and C. Martel. Analyzing and Characterizing Small-World Graphs. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, 2005. To appear.
- [83] P. Erdős and A. Rényi. On Random Graphs I. *Publicationes Mathematicae Debrecen*, 6:290–297, 1959.
- [84] C. G. Plaxton, R. Rajaraman, and A. W. Richa. Accessing Nearby Copies of Replicated Objects in a Distributed Environment. In *Proceedings of the 9th annual ACM symposium on Parallel algorithms and architectures*, pages 311–320, 1997.
- [85] P.O. Seglen. The Skewness of Science. *Journal of American Information Society*, 43:628–638, 1992.
- [86] R. Monasson. Diffusion, Localization and Dispersion Relations on Small-World Lattices. *European Physical Journal B*, 12:555–567, 1999.
- [87] S. Ratnasamy, P. Francis, M. Handley, and R. M. Karp. A Scalable Content-Addressable Network. In *Proceedings of the ACM SIGCOMM 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pages 161–172, 2001.
- [88] S. Ratnasamy, M. Handley, R. Karp, and S. Shenker. Topologically-Aware Overlay Construction and Server Selection. In *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies*, pages 1190–1199, 2002.

REFERENCES

- [89] R.V. Renesse, K. Birman, A. Bozdog, D. Dimitriu, M. Singh, and W. Vogels. Heterogeneity-Aware Peer-to-Peer Multicast. In *Proceedings of the 17th International Symposium on Distributed Computing*, pages 175–186, 2003. Available at <http://www.cs.cornell.edu/home/rvr/papers/Heterogeneity.ps>.
- [90] M. Ripeanu, I. Foster, and A. Iamnitchi. Mapping the Gnutella Network: Properties of Large-Scale Peer-to-Peer Systems and Implications for System Design. *IEEE Internet Computing Journal*, 6(1), 2002. Available at <http://people.cs.uchicago.edu/~matei/PAPERS/ic.pdf>.
- [91] A. Rowstron and P. Druschel. Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems. *Lecture Notes in Computer Science*, 2218:329–350, 2001.
- [92] R.V. Kulkarni and E. Almaas and D. Stroud. Exact Results and Scaling Properties of Small-World Networks. *Physical Review E*, 61:4268–4271, 2000.
- [93] S. Milgram. The Small World Problem. *Psychology Today*, 2:60–67, 1967.
- [94] Seti@home. The Seti@home project web site. <http://setiathome.ssl.berkeley.edu/>. 2003.
- [95] K. Sripanidkulchai, B. Maggs, and H. Zhang. Efficient Content Location Using Interest-Based Locality in Peer-to-Peer Systems. In *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies*, pages 2166–2176, 2003.
- [96] I. Stoica, D. Adkins, S. Zhuang, S. Shenker, and S. Surana. Internet Indirection Infrastructure. In *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pages 73–86, 2002.
- [97] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan. Chord: A scalable Peer-to-Peer lookup Service for Internet Applications. In *Proceedings of the ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, pages 149–160, 2001.
- [98] S.H. Strogatz. Exploring complex networks. *Nature*, 410:268–276, 2001.

REFERENCES

- [99] C. Tang, Z. Xu, and S. Dwarkadas. Peer-to-Ppeer Information Retrieval Using Self-Organizing Semantic Overlay Networks. In *Proceedings of the ACM SIGCOMM conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, pages 175–186, 2003.
- [100] H. Štefančić and V. Zlatić. Preferential Attachment with Information Filtering - Node Degree Probability Distribution Properties, Preprint, arXiv:cond-mat/0404495. Available at <http://www.arxiv.org>.
- [101] J. Zeng and W.-J. Hsu. Near Optimal Routing for Small-World Networks with Augmented Local Awareness. *ACM Computing Research Repository*, arXiv cs.DM/0409017. Available at <http://www.arxiv.org>.
- [102] J. Zeng and W.-J. Hsu. Optimal Routing in a Small-World Network. *Submitted*. Available at <http://www.cais.ntu.edu.sg/~zjy>.
- [103] J. Zeng and W.-J. Hsu. ReCord: A Distributed Hash Table with Recursive Structure. *ACM Computing Research Repository*, arXiv cs.DC/0410074. Available at <http://www.arxiv.org>.
- [104] J. Zeng, W.-J. Hsu, and Y. Hu. Looking Further Ahead Reveals An Even Smaller World. Technical Report CAIS-TR-2004-003, Centre for Advanced Information Systems, School of Computer Engineering, Nanyang Technological University, 2004.
- [105] J. Zeng, W.-J. Hsu, and S. Zhou. Construction of Scale-Free Networks with Partial Information. In *Proceedings of the Eleventh International Computing and Combinatorics Conference*, 2005. To appear.
- [106] H. Zhang, A. Goel, and R. Govindan. Using the Small-World Model to Improve Freenet Performance. In *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies*, pages 1228–1237, 2002.
- [107] B. Y. Zhao, J. D. Kubiatowicz, and A. D. Joseph. Tapestry: An Infrastructure for Fault-tolerant Wide-area Location and Routing. Technical Report UCB/CSD-01-1141, UC Berkeley, April 2001.