Interaction of Overlay Networks: Properties and Control

JIANG Wenjie

A Thesis Submitted in Partial Fulfillment

of the Requirements for the Degree of

Master of Philosophy

in

Computer Science and Engineering

Mar 2006

Thesis/Assessment Committee

Professor Michael Rung Tsong LYU (Chair)

Professor John C.S. LUI (Thesis Supervisor)

Professor WONG Man Hon (Committee Member)

Professor Don Towsley (External Examiner)

Abstract

Although the concept of application layer overlay routing has received much attention lately, there has been little focus on the "co-existence" and "interaction" of overlays on top of the same physical network. In this thesis, we present some fundamental insights of these overlay interactions. First, we show that when each overlay performs the overlay optimal routing so as to optimize its own performance, there exists an equilibrium point for the overall routing strategy (in other words, individual routing optimization will *not* lead to routing instability). However, the equilibrium point may be *inefficient*. We discover some implications due to the interactions: (a) the equilibrium point may not be Pareto optimal, (b) some fairness anomalies of resource allocation may occur. We further show the "performance gap" to the global optimal routing policy, which is often used by an ISP to control the underlying traffic. This is worthy of attention since overlay may not be aware of the existence of other overlays and they will continue to operate at this sub-optimal point. Thirdly, we explore two distributed pricing schemes to resolve the above issues. We show that by incorporating a proper pricing scheme, we can lead the selfish behavior of overlays to an efficient equilibrium. Extensive fluid-based as well as packet-based simulations are carried out to support our theoretical claims. We believe this work provides a fundamental understanding of overlay interactions, and stimulates further research on this issue.

摘要

尽管近年来应用层的 overlay 路由的概念已经得到了很多关注，然而对于多个 overlay 网络存在于同一个物理层网路的研究却很少。在这篇论文中，我们尝试对于多个 overlay 网络之间的相互影响做基本的研究分析。首先，我们展示当每个 overlay 网络采用最优化路由来提高自己网络的性能时，整个网络的路由策略将会收敛到一个平衡点，也就是，独立的路由优化并不会导致全局网路的不稳定。尽管如此，这个路由博弈的纳什平衡点不一定是最优的。我们同时发现了关于 overlay 间影响作用的一些负面效应。第一，路由平衡点不是帕雷托最优的；第二，网络资源分配的公平性无法得到保证，甚至导致出现一些矛盾的情况。其次，我们探讨在这个路由博弈中性能的损失，比较于因特网络运行商经常采取的一种全局最优的路由策略。我们的研究发现具有重要意义，因为在路由博弈的纳什平衡中，每个 overlay 网络无法得知其他 overlay 网络的存在，并且将持续在一个低效的平衡点运行。在下一步中，我们采用两种不同的分布式记价系统来解决以上提到的问题。我们证实当网络中采取适当的记价时，每个 overlay 自私的行为可以被引导至一个有效的平衡点。我们分别用基于流和基于包的网络仿真来验证我们的发现。我们相信此项工作对相关课题的研究迈出启发性的一步，并对未来的相关研究起指导性作用。

Acknowledgement

First and foremost, I am deeply indebted to my advisors, Professor John C. S. Lui and Professor Dah-Ming Chiu. Their encouragement, support and advice have been immensely valuable, both in personal and professional terms. I am particularly grateful for their emphasis on persistence and passion in research, and for the characteristics they have shown as an example for my development as an academic.

I am also grateful to my external marker Professor Don Towsley for his enthusiasm and counsel throughout my graduate career. It is his seminal work that motivates my inspiration of this work.

Special thanks go to my thesis committee members, Professor Michael Rung Tsong Lyu and Professor Man Hon Wong. Each devoted significant time and effort to my thesis, and their suggestions and comments led to substantial improvement in the final product.

The Advanced Networking and System Research Group in CUHK provides me an ideal environment to think, discuss and learn. I am particularly grateful to our study group, where I enjoy the course of learning and benefit from the collaboration with other group members.

I also cherish the time I spent with my colleagues and friends at CUHK. I would especially give my thanks to Sam Lee, T.B. Ma (Little White), Haibin Sun, Bin Fan, Yan Gao and Jessie Hui Wang, who shared their interesting discussions and insightful suggestions, as well as the very moment we spent together in this beautiful campus.

Lastly, I owe a great debt to my family, for their support and love, evermore.

This work is dedicated to my Mother and Father.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

The foremost design philosophy of the Internet is to build a highly scalable, simple and, hence, evolvable network. To achieve scalability, hierarchical routing is adopted and routing policies are left to network operators, e.g., Internet Service Providers (ISPs). Usually, non-load-dependent routing (i.e., link costs are kept constant) is used to keep things simple. When traffic demand changes, congestion control at the source regulates the traffic load to provide a "best effort" service rather than tries to find an alternate path which has a higher available network bandwidth or better performance. Although such design choices might not achieve the optimal performance, they are considered (and well deservedly so) as the contributing factors to the scalability and robustness, hence the success of Internet [1].

This form of simple but sub-optimal network design leaves the door open for more intelligent traffic routing at the application layer. For the past few years, there have been tremendous interests on the routing and deployment of overlay or peer-to-peer networks [2,3]. The overlay architecture provides a possibility for the end users/hosts to deliver their information through peers or overlay nodes to the ultimate destination. Therefore, such system design possesses a degree of freedom in choosing the logical level overlay routes, which allows applications to control the performance of their routes and fully explore the availability of the Internet resources. While the success of overlay networks attributes to their flexibility and efficiency that traditional IP-level routing cannot achieve, it is also essential due to its simplicity since there is no need to modify the existing IP infrastructures. In some recent work, the concept of overlay network is proposed as a promising approach to enhance quality-of-service (QoS) in today's Internet [4, 5].

As one can witness the growing trend of setting up P2P and overlay networks, some examples include content delivery networks like Akamai [6], resilient overlay networks like RON [2], end system multicast like [5], cooperative file storage system like Chord [7] and application layer P2P file distribution like BitTorrent [8]. In particular, application-layer routing schemes are shown to effectively address the problems of traditional IP routing. For example, results from [2] demonstrate that most of the time, one can find a better route to ensure QoS guarantee service. Also, measurements from [9, 10] indicate that in the current Internet, a large percentage of flows are able to find a better route by relaying packets with the assistance of overlay nodes, and thereby improve their performance. From a theoretical point of view, application routing is a form of *optimization* in which the overlay tries to maximize its desired objective, such as reducing the delay and/or cost, subjected to the availability of network resources.

1.2 Challenges

Overlay network is not a new concept. The Internet can be viewed as an overlay network, built largely on top of telephone and other (e.g. frame-relay, ATM and others) networks. The Internet's underlay provides specific bandwidths, therefore emulates physical wires. When building overlays on top of the Internet, however, the logical links between nodes are based on Internet's best-effort service. Consequently, the performance of an Internet overlay will depend on how it co-exists with (a) the existing Internet traffic, and (b) the traffic of other Internet overlays sharing the same physical links.

For traditional overlay routing, the strategy is to select the *best* path for a given flow so that the performance, say end-to-end delay, can be minimized. Note that this form of overlay routing does *not* split the flow among all available paths, but rather selects a path which has the minimum delay. Because the existence of this flow will increase the congestion level of the traversed links, the average delay of this flow is generally not minimized. Furthermore, it is pointed out in [11] that there is a global performance degradation due to this form of routing. In [12], authors present an architectural framework of shared routing underlay so as to offer a query service for all overlays above the physical network. For instance, an overlay can determine the congestion level or delay of a physical link. In [13–15], authors show that when overlays have the ability to assign traffic among its available paths, it can minimize the average delay. These results suggest the feasibility of implementing an optimal routing scheme on an overlay network.

Although the concept and deployment of application layer overlay networks have received much attention recently, there has been little focus on (a) the "*co-existence*" and "*interaction*" of multiple overlays, (b) the "*control*" of this interaction so as to ensure overall network performance and/or fairness. In this work, we consider scenarios in which multiple overlays are constructed on top of a physical network. These overlays may share some physical links or nodes, but they may not realize the existence of other overlays. Each overlay is "*selfish*" by nature in that it performs overlay routing so as to optimize its own performance without considering the impact on other overlays. Therefore, there is an inevitable interaction between these overlays due to their individual optimization actions. However, some important questions need to be addressed:

- What is the form of interaction?
- In which ways can the interaction affect the network stability, performance as well as fairness in the resource allocation?
- When resource competition is unregulated, what is the price of anarchy?

1.3 Our Contribution

In this work, we seek to understand the fundamental properties of Internet overlay interactions systematically. We analyze the behaviors of overlay networks under a game-theoretic framework. We demonstrate the existence of a Nash equilibrium in a routing game in which all overlay participated. Implication of the equilibrium shows that the interaction has some intrinsic properties as well as some undesirable effects (i.e., degradation of overall network performance, fairness anomaly in resource allocation, which we will present in Chapter 7). To control these effects, a pricing scheme is proposed to reduce or eliminate these effects. Note that the pricing scheme can be used by the ISPs, not only to increase their revenue, but also as a means to perform traffic engineering so as to achieve the global optimality, or improve fairness.

In summary, the contribution of our work is as follows:

- We present and formalize the "*overlay optimal routing*" policy in an optimization framework. We show how each overlay can solve this routing optimization problem distributively.
- We model the interaction of overlays as a non-cooperative game and show that there exists a Nash equilibrium (i.e., implying that there will be no routing instability) under a general network setting.
- We show that an overlay can choose to achieve three different levels of

optimality: user (selfish) optimality, overlay optimality and global optimality (each requiring different levels of knowledge about the network). We show that by adopting different routing strategies, the system can achieve a spectrum of performance balanced between efficiency and fairness.

- We report a number of anomalies due to interactions between overlays. Firstly, the equilibrium point is not Pareto (or social) optimal, which can cause a "tragedy of the commons", meaning that the performance of *all* overlays can be seriously degraded. Secondly, an interesting and more important discovery is that the interaction may lead to some fairness anomalies in resource allocation. That is, at the equilibrium point, it is possible for some overlays to obtain a higher percentage of the common resource (e.g., link bandwidth) as compared to other overlays, and cause these overlays to experience a significant performance degradation.

- To overcome the above two issues, we propose two pricing mechanisms to alleviate the problem. We show that by incorporating a proper pricing scheme, we can either improve the overall performance, or bring in the fairness in resource allocation, and at the same time, increase the ISP's revenue.

- We illustrate via fluid-based and packet-based simulations to support our theoretical findings.

1.4 Structure of the thesis

The balance of the thesis is as follows. In Chapter 2, we present a brief introduction of overlay networks and different overlay routing policies. In Chapter 3, we formalize the overlay optimal routing policy and its mathematical optimization model. Simple examples are provided to illustrate the model. We then study the interaction between co-existing overlays as a non-cooperative Nash routing game and show the existence of a Nash equilibrium in Chapter 4. In Chapter 5, we compare the performance and fairness indices for different overlay routing policies. In here the performance gap and the price of anarchy are further demonstrated. Fluid as well as ns-2 simulations are performed to show the properties of the equilibrium point in Chapter 6. In Chapter 7, we illustrate when multiple overlays use this form of application routing, interaction occurs and there exist some potential problems on performance and fairness. In Chapter 8, we present two pricing schemes to resolve the above mentioned anomalies. Related work is given in Chapter 9 and finally Chapter 10 concludes.

Chapter 2

Background Study

Over the past few years, we have seen an emerging trend in setting up various types of peer-to-peer or so called "*overlay networks*". There are numerous examples of such overlay networks including content delivery networks implemented by companies like Akamai [6], resilient routing overlay networks like RON [2], end system multicast like [5], application layer P2P file distribution like BitTorrent [8], and research testbed networks such as PlanetLab [16]. These overlay networks provide an opportunity to enhance the existing functionalities supported by the Internet infrastructures. The appearance of overlay networks blurs the boundary between what we think of as the "Internet", and the applications that sit on top of it. Overlay networks also blur the boundaries between the concept of "network edges" (what we think of as the customers such as computers and end hosts), and "network cores" (what we think of as the servers and routers). As such, the technological and policy implication reflected by the overlay networks has a profound impact on the evolution of the Internet architecture, disrupting the traditional "end-to-end" design principle [1, 17].

Since overlay networks are becoming one of the most prominent areas in research and industry development, we need a clearer understanding of what constitutes an overlay, the motivation for its implementation and use, how it works and their potential conflicts and interactions that may arise as they evolve. This chapter provides an attempt to define overlay networks. We then introduce a taxonomy of these overlays with some examples. In particular, we are interested in the routing overlay networks such as RON and analyze their routing policies.

2.1 An Introduction to Overlay Networks

During the last few years, overlay networks have become one of the most prominent tools in research and Internet industry. The appearance of overlay networks allows system designers to perform their own routing and traffic managements to assist specific applications on top of the existing Internet infrastructures. The Internet itself began as an overlay network, built largely on top of the public-utility regulated telephone networks, using long-distance telephone links to connect Internet routers. The evolution of the Internet undertook a process of complementing the underlying infrastructures by adding new functionalities (packet-switch data communication) to support new communication requirements (end-to-end computer communication). The success of the Internet owes much to the inter-operability and connectivity supported by ubiquitous adoption of the IP protocols and adherence to the end-to-end design philosophy that has governed the growth of the Internet. Also with growth comes the new requirement of heterogeneous services (differentiated quality-of-service needs for different applications) as well as a solution to support real-time applications and enhanced security. Hence, overlay networks are born in this new epoch to meet the challenges stated above.

2.1.1 What is an Overlay Network?

Much alike the Internet emulates the physical wires based on the telephony underlay, a modern overlay network operates similarly, using the Internet paths between end hosts as virtual "links" upon which the overlay routes data and builds up a network. As a result, an overlay can be leveraged to deploy new functionalities and services almost immediately, without years of upgrading Internet routers to be equipped with devices that can perform application-specific packet handling and forwarding. It well presents developers with a flexible and powerful platform on which to provide new services such as multicast, anycast and mobility [18, 19].

While overlay networks are at least as old as the Internet, they were not generally regarded as an area of research until the late 1990s, when two types of overlays become popular: *routing overlay networks* and *content delivery networks* (storage and look up overlays).

As a starting point, we first offer a definition of an overlay network given in [20] as follows:

An Overlay is a set of servers deployed across the Internet that

1. *provides some sort of infrastructure to one (or ideally several) applications,*

2. *in some way takes responsibility for the forwarding and handling of application data in ways that differ from or are in competition with what is part of the basic Internet.*

3. *is operated in an organized and coherent way by a third party (which may include collections of end-users) to provide a well-understood service that is infrastructure-like, but*

4. *is not thought of as part of the basic Internet.*

To fully understand various dimensions of this definition, it is helpful to look at an overlay network from multiple perspectives: *architecture, functionality* and *commercial policy.* Some elaboration is made. We are going to show how the overlay's architecture is embedded into current Internet, the basic functionalities supported by various types of overlay networks and how overlays relate to the industry structure.

Architecture

First of all, the definition of an overlay network is that of a "*logical*" network "*on top*" of the basic Internet, utilizing the Internet infrastructures, while providing "*infrastructure-like*" services to the upper layer applications that run on the overlay nodes. From a perspective of the OSI Reference Model, an overlay can be viewed as a middle layer that sits above the IP layer but below the application layer. However, this definition is not precise. From another perspective of the evolving architecture of the Internet, we should also look into how overlay networks relate to the end-to-end design philosophy that is considered as one of the contributing factors to the success of the Internet.

In general, the Internet, in its simplest form, consists of two components: (1) *end nodes*, which include computers at the edge of the network that play the role of servers, work stations, ordinary users and so on; (2) *routers*, which store-and-forward packets for end nodes. Therefore, the Internet is often regarded as a cloud of routers that inter-connect groups of end-users at the edge of the information world. Under this framework, the boundary between end-users and routers is quite clear. On one hand, applications such as games, media streaming, etc do not run on routers and routers know nothing about the specific functionalities and QoS requirements of different applications. The only job of a router is to forward packets to the destination. On the other hand, all communication is performed in an end-to-end manner. End users or computers behave by sending or receiving packets but do not "deliver information" (or forward packets) as a router does. However, the advent of overlay networks disrupt the distinction between the two concepts. As for an overlay node, it is hard to say whether it is a router or an end-user. First, overlay nodes are distributed around the Internet in a way that provides an infrastructure on which the application runs, from which perspective they are end users. Second, overlay nodes handle and forward application data for their peers, performing a task that is usually done by routers but more *intelligently*. To see why overlay networks allow a more intelligent routing scheme, let us further look into the extra functionalities that overlay networks can provide and their benefits to end users.

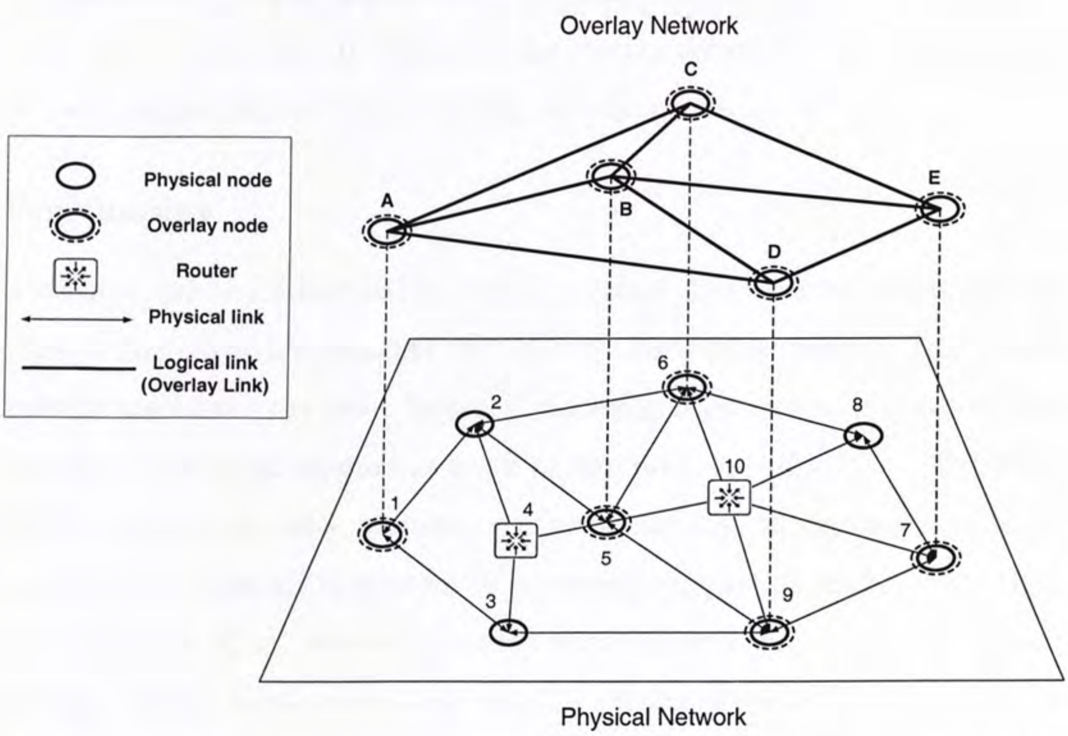


Figure 2.1: An illustration of an overlay network on top of the underlying physical network

Figure 2.1 illustrates an example of an overlay network on top of an underlying physical network. Note that the overlay nodes cooperate to form a logical topology. Depending on the functionality and different types of overlay networks, the logical topology can be organized in a variety of ways. Figure 2.1 shows one possibility. By leveraging the routing functionality provided by the overlay networks, one overlay node can utilize multiple available paths with the assistance of other overlay nodes forwarding packets for it. This greatly improves the flexibility and robustness of the routing performance, compared to the underlying policy-based IP-level routing. It is interesting to note is that the logical link between two neighboring overlay nodes is a physical path, which is still determined by the IP-level routing. For example, consider the logical link BE, it is comprised of a set of physical links $5-10$ and $10-7$. Hence, two different logical links may share some common physical links. The physical link $1-2$ shared by two logical links $AC = \{1-2, 2-6\}$ and $AB = \{1-2, 2-4, 4-5\}$ is an obvious example. In this sense, an overlay network is infrastructure-like but still dependent on the underlying infrastructures.

Functionality

A second way to understand an overlay network is to ask questions such as: what is the extra functionality provided by an overlay network that distinguishes itself from the basic Internet? In the context of our discussion here, we refer to the basic Internet as a suit of core protocols (IP, TCP, UDP, BGP, OSPF) that constitute a minimal set of protocols that all networks and nodes must support necessarily in order to be considered part of the Internet. However, the "best effort" services provided by the basic IP protocols is not always enough. Many applications have specific quality-of-service requirements in route selection and more general communication needs, i.e., multicast, security, content distribution and mobility. To provide these services, not provided by the underlay network, overlays appear as a solution that blurs the clean Internet architecture distinction between packet forwarding and application processing. In a later section, we are going to present a taxonomy of various functionalities supported by different types of overlay networks.

Commercial Policy

As noted earlier, the Internet began as an overlay on top of telephone networks which are regulated by the government. Because of the limited services telephone companies can provide, public policy and industry dynamics gave rise to an ISP industry. These service providers leased the physical infrastructures from telephone companies and combined them with packet switching technology to support the Internet. Hence, from an industry perspective, the basic Internet infrastructures should be defined as the services that are provided by the ISPs. As one might guess, the emergence of overlay networks blurs the concept of an ISP. Usually, overlay services are typically not provided by traditional ISPs but by *third parties.* They purchase the basic Internet services, such as connectivity and access to the Internet, from the traditional ISPs and resell the services combined with application specific features. Sometimes, this type of overlay networks is also referred to as the "*virtual ISP*" [21, 22]. Akamai, a provider of CDN services, is an obvious example. Another type of third-party overlay networks is comprised of a group of end-users as is the case with peer-to-peer networks. Many interesting and emerging overlays (peer-to-peer, routing) are first deployed by edge users in end-nodes and may not generally be thought of as infrastructure providers.

2.1.2 Benefits of Overlay Networks

There is a growing feeling among many Internet researchers that the Internet protocol (IP) and the IP routing infrastructure has become ossified by virtue of its huge success. Changing the hundreds of millions of currently deployed IP-speaking devices poses a considerable challenge. The deployment of IPv6 is an obvious example. Overlay networks offer an alternative to modifying Internet protocols or routers, providing a quick and easy deployment path that lacks many of the technical and political hurdles of a router-level deployment.

Instead of changing the IP layer, many researchers now design protocols that run on top of IP in an overlay.

Another benefit of overlays is that they avoid burdening the underlying network with features better performed at higher layers. For example, content routing requires that the content routers "understand" the application protocols that run through them. Augmenting core routers with application-specific knowledge would burden them with processing needed only by a small fraction of the traffic that passed through them, and force them to undergo frequent updates.

Finally, an overlay can take the advantage of the large glut of processing, memory, and permanent storage available in commodity hardware to perform tasks that would ordinarily be well beyond the ability of a conventional router, such as expensive cryptographic operations, file caching, or database lookups. Performing these slow and expensive tasks in an overlay keeps them off of routers' critical paths. The ability to perform these tasks enables the creation of powerful new facilities such as scalable, distributed publish-subscribe systems and content distribution networks.

In summary, we offer a list of justifications why overlays emerge as the Internet itself is becoming more persuasive:

- First, overlays may exist to support the special requirements of a particular class of application or user community. If the needs of a particular user community differ from those of the general Internet, people may seek to address the needs through specialized capabilities that are separate from but work in conjunction with the basic Internet. Thus, the success of Internet as an open platform leads to the need to satisfy the heterogeneous requirements, providing one justification for an overlay network.

- Second, overlays may play a role in the dynamic evolution of the Internet technology. The very success of the Internet is due to the ubiquitous adoption of the end-to-end architecture and IP protocols, which also poses a challenge when it comes to upgrading the Internet's own basic infrastructures. Coordinating the updates of all routers and servers represents a massive undertaking. Therefore, overlay networks are a feasible solution.

- Third, overlay networks present a first experiment to add in new functionalities of current routing and architecture design. By incrementally deploying new solutions, we may provide services to those users that most require enhancements which may not be available yet in the current Internet. Examples include enhanced quality of service (e.g., reduced delays from better routing) or security/privacy (e.g., onion routing to protect identity).

- Finally, overlays may arise because of conflicts in the interests of different Internet entities, reflecting a tussle between and among customers, service providers, and policy-makers. For example, routing overlay networks seek to improve the route selection process, which may be in conflict with policy-based routing implemented by peering ISPs with non-delay-related considerations. Hence, an overlay that tries to select the "best" route based on global information about link delays may violate business agreements about traffic routing between ISPs that are seeking to manage traffic to minimize intercarrier payments. Related work can be found in [23–25].

2.2 Taxonomy of Overlay Networks

In this section, we provide a taxonomy of different types of overlay networks. Depending on their functionalities, overlay networks can be categorized into three types: Content Delivery Networks (storage and lookup overlays), Routing Overlays and Security Overlays. Table 2.1 summarizes a list of overlay networks categorized by their functionalities and purposes. We examine the properties of these overlay networks by their organizations, technical and policy challenges. Note that we do not attempt to present a detailed description of the architecture and its technical implementation, but rather to offer the readers a very rough idea of their common characteristics.

2.2.1 Routing Overlay Networks

The first type of overlay networks we are going to examine is Routing Overlay Networks, which is our primary focus in this thesis. A routing overlay network is an overlay that exists for the purpose of controlling and modifying the path of data packets through the network. Different from content delivery overlays and security overlays, the information exchange between two end-points in a routing overlay remain the same as what they would have been in the absence of the overlay. However, the route through which the data packets are forwarded between the two end-points may differ.

Routing overlay networks are unique among classes of overlay networks that we discuss in this thesis in that they implement some functionality that is already provided by the basic Internet infrastructure. In contrast to other classes of overlay networks which provide new functionalities, routing overlay networks are designed to enhance or "*override*" the existing routing function. It is this overlap, between routing in the underlay Internet and routing as an overlay function, that leads to the most interesting properties of routing overlays. Furthermore, as the issue of routing interactions between overlays

Category	Functionality & Purpose	Example
Peer-to-Peer (p2p)	File sharing & distribution (video, mp3s)	Napster [26], Gnutella [3], BitTorrent
Content Delivery Network (CDN)	Harnessing the power of large, distributed content caching, to reduce access delays and transport costs	Akamai, Chord [7], Pastry [27], Tapestry [28], CAN [29], SkipNet [30]
Routing Overlay	Reduce routing delays, resilient overlay routing, enhance or replace IP-routing, provide new functionality or improved services	Resilient Overlay Networks (RON), Akamai SureRoute, Tor (onion routing) [31], Project IRIS [32]
Security Overlay	Enhance end-user security and privacy	SOS (Secure Overlay Services) [33], Virtual Private Networks (VPN), onion routing (Tor, I2P), anonymous content storage (Freenet [34], Entropy [35]) censorship resistant overlays (Publius [36], Infranet [37], Tangler [38])
Experimental (research) Overlay	Facilitate innovation, implementation of new technologies, experimentation	PlanetLab, I3 (Indirect Internet Infrastructure)
Multicast Overlay	Implement end systems with multicast related functionalities, i.e., membership management and packet replication	End System Multicast [5], Mbone [39], 6Bone, TRIAD, IP-NL
Others	Various Purposes	VoIP (Skype) [40], Delay Tolerant Networks [41]

Table 2.1: Examples of Overlay Networks

themselves is brought to discussion, we are going to see some fundamental yet profound implications that require study, which constitutes the motivation of our work in this thesis.

Underlay Routing in the Internet

Before proceeding to an introduction of routing overlays, it is helpful to have a brief discussion on the routing function performed by the existing Internet, in order to obtain a clear understanding of *why* routing overlays emerge.

First of all, we provide the definition of "routing", which is one of the very basic functionalities supported by the Internet. Routing, is the determination of a path between the source and the destination for data transmission, the basic function of all computer networks. It is trivial if there is only one possible route between the source and the destination. Any communication must follow this path and otherwise communication fails. However, for any reasonably large network, routing is more complicated since there are multiple possible paths between any given source-destination. In this sense, routing becomes an *optimization* problem in which the network operators seek to choose an *optimal* path. As we will see, even in a single network, the definition of "optimum" is not obvious and involves some mixed considerations of ISPs who manage the network.

From the ISP's perspective, routing becomes a set of distributed optimization problems with different objectives. Since there is no universal criteria to determine which path is the *best*, there is no omniscient observer to choose a *globally* optimal path. Instead, the path taken by a packet is a result of individual decisions made by each of the many ISPs that combine to form the Internet. These routing decisions are driven by a number of factors. Chief among these are the internal structures of ISPs, which include the operational cost of a packet from when it arrives at the ISP to when it leaves, and the business agreements between ISPs and its peers, which determine the cost to deliver the packet to the next ISP. These individual considerations are co-ordinated under a network protocol known as the Border Gateway Protocol (BGP). Generally speaking, BGP allows each ISP to express explicitly its policies for accepting, forwarding and receiving packets. By considering all these policies BGP computes in a distributed manner the *best* path along which the packet is forwarded.

However, the obscure definition of "best" also introduces some difficulties. The notion of "best" is not sufficient to fully express the routing task, since best is a single dimensional concept while routing is a multi-dimensional problem. That is, routing as a decision co-determined by many ISPs, reflects a variety of purposes sought by different ISPs. Among these objectives might be:

1. The cost of a packet along its path.

2. Traffic engineering task performed by ISPs to balance the distribution of traffic along different physical links, such that the utilization is maximized or the congestion level is minimized.

3. Performance, such as available bandwidth or end-to-end delay for some traffic across the ISP.

Therefore, the routing decision as a result of individual optimizations with different objectives, may not capture the single notion of "optimality".

In summary, we draw two conclusions about the current IP routing system. Firstly, the path over which data packets are routed is entirely determined by the ISPs, rather than the end-user or specific applications. Secondly, current route selection is a result of mixed optimizations towards a mixed objectives of cost, delay and operational efficiency, rather than any metric directly related to application performance. Hence, we now turn to the concept of routing overlay networks.

Routing Policies in Routing Overlay Networks

As previously stated, the basic functionality of a routing overlay network is to override the routing function already implemented by the Internet infrastructure. We illustrate the mechanism of overlay routing using a simple example depicted in Figure 2.2. Before proceeding to the introduction of overlay routing, we start with a brief overview of the impact of ISP policies on how packets are routed through ISPs.

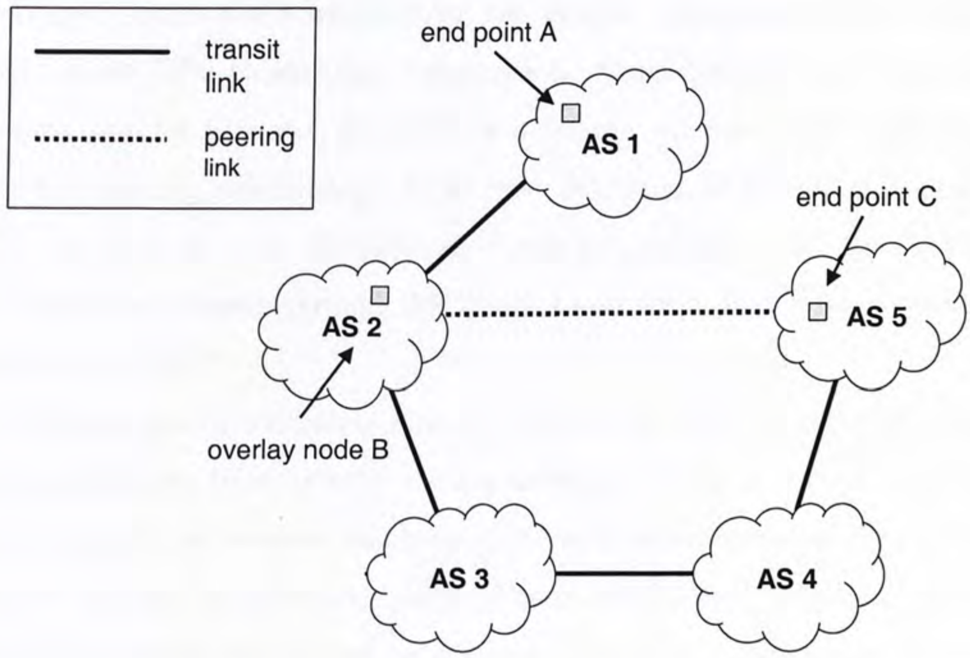


Figure 2.2: Example of overlay routing

The Internet is comprised of multiple autonomous systems (ASes) or ISPs as we referred to above. Generally speaking, each AS is controlled by an independent commercial entity. The connectivity between ASes generally corresponds to one of the two relationships: transit and peering. For instance, if there is a transit relationship between AS 1 and AS 2, then packets destined for nodes in AS 1 can be routed through AS 2, and vice versa. These routes are publicly advertised, and often these globally advertised transit relationships involve some financial compensation. On the other hand, if AS 2 and AS 5 have a peering relationship, then customers of AS 2 and AS 5 can reach each other through the peering link at no charge. However, these peering links are not publicly advertised, and are generally considered confidential. As a result, other ASes cannot route their packets destined for AS 5 through AS 2, and destined for AS 2 through AS 5 as well. As only customers of AS 2 and AS 5 can send packets through this peering link, this peering relationship is often agreed upon by the two parties and is reciprocal to both.

Transit routes are advertised by the Border Gateway Protocol (BGP), which allows ISPs to exchange information. Consequently, an AS can find the next hop for a packet destined for a remote AS with which there is no transit or peering relationship. As we have discussed, BGP cannot guarantee to find an optimal path. As shown in Figure 2.2, a packet going from AS 1 to AS 5 needs to traverse through AS 2, AS 3 and AS 4, if it follows the route advertised by BGP.

Now suppose the end-points A and C, situated within AS 1 and AS 5 respectively, participate in an overlay routing network. Without loss of generality, each end-point can be seen simply as a PC with software installed associated with the overlay, or with some abuse of term, might be a corporate, government or university location where a server runs software associated with the overlay on behalf of users within the location. Suppose the overlay has a node B in AS 2, and, though not relevant for the example presented here, might also have nodes in other ASes as well. Therefore, if the packet is first sent from A to B, then it could be sent from B to C by taking the advantage of the peering relationship between B and C. This indicates that using application level intermediate nodes could provide a transmission with a more efficient path.

However, the motivation for routing overlay networks is not simply circumventing the financial relationships of ISPs. In particular, through some measurements, if endpoint A is aware of the current performance parameters, such as loss and latency, of the BGP-based route and the overlay route, it may choose either of the routes depending on the achieved performance. Further, it might switch back and forth, using each route for a period of time when it is preferable. As previously stated, the BGP protocol is built from the ground up to be highly scalable and applies some policy constraints, however, it does not necessarily result in optimal paths. Additionally, the BGP routing protocol does not probe paths for performance so it is unable to detect and route around congested links. Recent studies in [42,43] have documented these characteristics. Furthermore, studies in [44] discovered that due to implementation issues and ambiguity with the protocol, the convergence time for BGP routing tables to stabilize after a link or other failure is on the order of many minutes. In addition to the problems with BGP, some rising incidence of major routing pathologies and path outages is documented in [43]. Consequently, end-to-end routing mechanisms neither guarantee the best path between end hosts, nor do they recover quickly from link failures. All these factors constitute the reason for the rise of routing overlay networks.

Technical Implications of Routing Overlays

Early research on routing overlays focused on their ability to improve application performance by selecting a higher-quality path through the network than what is selected by the BGP protocol. Experimental results reported by a number of researchers have demonstrated this capability in practice. With overlay routing, end-users can potentially attain lower latency, lower loss, higher throughput and increased availability. In particular, the merits of routing overlays can be shown in the following aspects:

- The route selection in an overlay network is intrinsically tuned to the specific needs of different applications, rather than relying on the generic route chosen by BGP. This property is essential since in most of the cases, the generic route is not optimized in terms of application performance.

- Overlay routing is able to fully utilize the availability of network resources, selecting non-default routes with low load, hence achieving minimal congestion delay.

- Overlay routing is able to respond quickly to the failures in the network by choosing alternative functioning paths. This greatly improve the robustness of the system.

- Overlays can "work around" the effects of ISP load management and traffic engineering, but at the same time, ignore and violate policy constraints.

Though the merits of routing overlays seem appealing, it is still too early to widely deploy them before we obtain a clear understanding of the negative technical effects coming along. To be precise, these effects are due to the uncoordinated control of routing by many different entities acting an optimization process *independently*. Among these effects are:

- First of all, severe performance degradation can occur if several overlays simultaneously shift their traffic from highly loaded path to a path with lower load. Since these activities are not coordinated, one possible effect would be to over-shit traffic, which results in congestion in the newly selected path. What's worse, this effect continues and all overlays end up in route oscillation, which severely undermines the system stability. Some recent studies on this issue were documented in [45–47].

- A second negative effect which is even more severe results from the interaction between overlay routing and traffic engineering performed by the underlying ISP. Studies in [48] suggest that sustained oscillation can occur if both layer respond to a disruption simultaneously, especially when the volume of overlay traffic is considerable.

- The third important yet not well addressed effect is due to the interaction between multiple co-existing overlay networks, which is the topic we are going to discuss in the remaining of the thesis. We show that some serious anomalies regrading the performance degradation and fairness paradox may continue to exist in an *inefficient* routing equilibrium.

Therefore, should routing overlays become widely deployed, it is essential for researchers to look into these problems and mitigate the negative effects with additional mechanism, such that we can achieve a coordination of the actions taken by multiple overlapping overlays. In a later chapter, we are going to show that our attempt to employ a pricing mechanism is a promising approach.

Implications of Routing Overlays on the Interests of ISPs

Implicit within the notion of application routing overlays is that control of the route selection is, at least to some extent, wrested away from the network operators and shifted to the end user. This loss of control over a basic function of network operators presents strong implications for the interests of the ISPs.

We are going to explore these implications using the same example illustrated in Figure 2.2. Suppose for any given destination in AS 5, AS 1 has chosen a BGP path via AS 2, AS 3 and AS 4. However, the end user A, who decides to use the overlay path between AS 2 and AS 5, has effectively overridden the ISP's decision. If the volume of overlay traffic is appreciable, i.e., many overlay users decide to utilize the overlay path and traffic is aggregated, the effect could be detrimental to ISPs for a number of reasons:

- First of all, overlay routing to a large extent shifts the traffic demand among different ISPs, which has an intense impact on the economic considerations of ISPs. To be concrete, AS 2 and AS 5 sign up a peering relationship for some commercial reasons, for instance, the volume of traffic from AS 2 to AS 5 and that in the reverse direction is somehow similar. However, the existence of an overlay path from AS 2 to AS 5 greatly increases the traffic volume in this direction, which breaks the balance of traffic flows in two directions. One immediate effect is the increased cost and traffic load on AS 2 (increased traffic demand because of a free peering link). AS 3 and AS 4, whose revenue are decreased, are also under the negative effects of overlay routing.

- From another perspective, AS 2 may make its choice based on some traffic engineering considerations. For example, to balance load on its links, (to achieve maximum efficiency and minimum congestion cost), AS 2 may apportion traffic between AS 3 and AS 5 based on historical traffic volumes. In this case, the benefit to some users of the overlay will be compensated for by a degradation of service to other users, and a loss of overall efficiency for the ISP. As the ISP tries to rebalance its traffic, an adverse cycle is likely to begin.

From this example we see that this interesting problem is due to the changing relationship between ISPs are their customers. Routing overlays change this relationship by giving the end users an input into the routing decision. As long as ISPs retain complete control over routing decisions within the network, these interactions will occur. To date, however, there is no coordinated way to resolve conflicting objectives between various parties. It is this lack of coordination that leads to many of the negative effects that overlays bring to us.

2.2.2 Content Delivery Networks (CDNs)

The second type of overlay networks we are going to examine is Content Delivery (Distribution) Networks (CDNs). Content Delivery Networks are overlays dynamically cache content and services at distributed locations throughout the Internet. They are interesting overlays to examine because they represent a large share of the overlay traffic on the Internet today and in the future, and are associated with commercial offerings from Akamai.

CDNs are technologically fairly straightforward. A CDN consists of a collection of non-origin servers that attempt to offload work from origin servers by delivering content on their behalf. The servers belonging to a CDN may be located at the same site as the origin server, or at different locations around the Internet, with some or all of the origin server's content cached or replicated amongst the CDN servers. For each application request for contents, the CDN attempts to locate a CDN server close to the client to server the request, where the notion of "close" may involve geographical, topological, or latency considerations. With content distribution, the origin servers have control over the content and can make separate arrangements with servers that distribute content on their behalf. CDNs are overlays because the IP layer is responsible for delivering the packet to the appropriate destination but the decision about the source of packets is made at the application layer by the redirector, and not the original requestor.

As we are going to explain below, CDNs emerged because of an unmet need of end-hosts for lower latency access and delivery of content, as well as a desire to reduce the transport costs of content and Internet service providers. At a technically level CDNs consist of caches of content and services distributed across the Internet. Figure 2.3 illustrates the architecture of a typical content distribution network. These caches contain copies of content and services retrieved either on-request or proactively from publishers and providers. The core of a CDN is the method by which requests and contents are routed and redirected in the overlays to accomplish the load balancing. There are two basic approaches to accomplish routing and redirection in distributed CDNs: DNS and URL rewriting, and http redirection. In balancing the request and content load, CDNs optimize different criteria including technical measures such as response time and server loads and economic measures such as bandwidth costs. A detailed description of these techniques are out of scope of this thesis and interested readers can refer to [49–52].

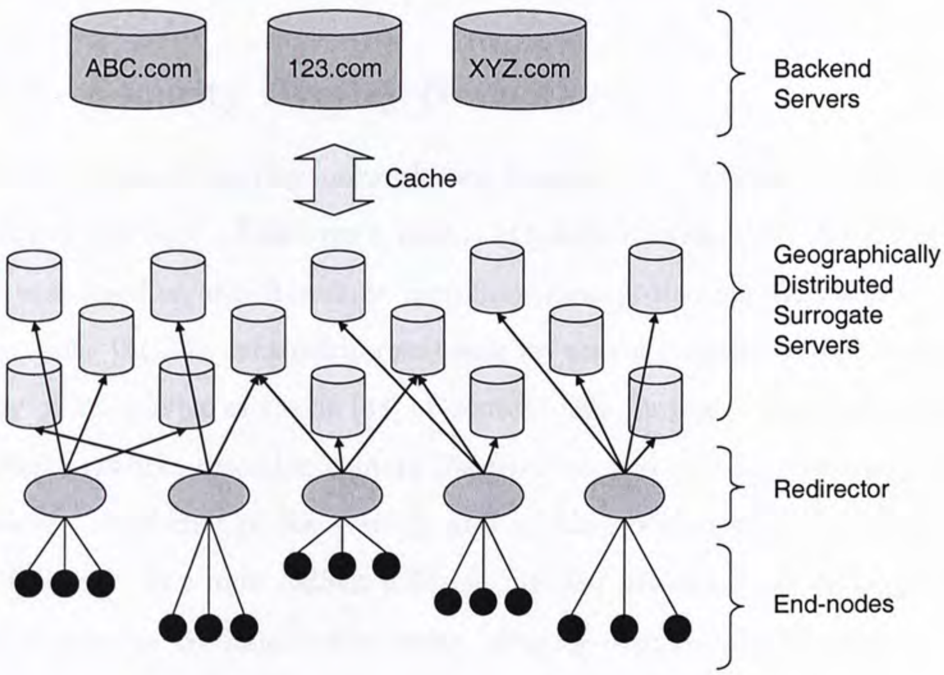


Figure 2.3: An illustration of content delivery networks

While technically straightforward, the impact of CDN overlays raises many interesting questions as CDNs evolve and grow. Unlike routing overlays which do not alter the communication pair addresses (source/destination), a CDN dynamically changes the communication pair by redirecting communications to different destinations. Thus, the CDN may be seen as an overlay infrastructure on top of the IP layer that supports multiple applications. Fundamentally, CDNs change the patterns of traffic on the Internet and more contents and services can be accessed locally, so they have a clear technical impact. Because traffic patterns also determine the money flows between providers, CDNs influence the commercial relationships on the Internet, which in turn, gives rise to policy implications. What's more, the growth of CDNs also has implications for capacity planning and where infrastructure investment is likely to occur. Hence, in the future, the evolution of CDNs may contribute to the creation of a two-class Internet, one that is high quality for commercial content, and on that is lower quality for non-commercial user-centric content.

2.2.3 Security Overlay Networks

The final class of overlay networks we examine are those we characterize as "security overlays". There are a variety of information security that these overlays seek to offer, which include communication protection [53], user or server anonymity [31, 34], censorship resistance for online content [37, 38] and deniability of knowledge of traffic [34] or content [36]. In many ways there security overlay networks resemble content distribution and routing overlay networks. However, they change the routing and caching behaviors of information on the Internet. The most salient difference is that instead of optimizing routing performance or communication costs, security overlays aim to enhance some aspect of end-user security. Some provide secret communication or anonymity for end users, while others make contents robust against attempts to remove it from the Internet and enable users to establish legal deniability of traffic or content ownership. Though the emergence of this type of overlays tend to increase the opaqueness of the Internet, they are well deserved to improve the user security significantly. In the following, we are going to present a very brief introduction of the mechanisms of security overlays. Note that we do not provide an exhaustive survey of this class of overlays, but provide a descriptive examination of each category instead. For those readers who are interested in this class of overlays, please refer to [54, 55].

The most popular type of overlay networks that provide a security property is the Virtual Private Network (VPN). VPNs provide encrypted tunnels between points in the network, extending the connectivity between networks across multiple geographic domains or between one's desktop computer in the office and a travelling laptop. Since the topics on VPNs have been well studied [56], we do not attempt to mention them further. The following list summarizes some examples of recently prominent security overlays:

- *Onion Routing Overlays*: This type of overlays enables anonymous communication over the Internet. Tor [31] and I2P [57] are current examples.
- *Anonymous Content Storage and Retrieval Overlays*: These overlays reveal the identity of authors, publishers and content providers when they store, query or download contents from the Internet. Examples include Freenet [34] and Entropy [35] projects.
- *Censorship Resistance Overlays*: This category of overlays seek to make it very difficult for those powerful adversaries to remove content or pollute the overlay networks with distracting contents. Examples include Publius [36], Infranet [37] and Tangler [38].

As one common characteristic of the above introduced security overlay networks is to harness the distributive power employed by the overlay technology, they are no different from content delivery overlays or routing overlays from a perspective of "tunnelling" architecture. Furthermore, we should also look into how these security overlays provide special functionalities to achieve user protection. To elaborate, we analyze the following three perspectives: how to provide *anonymity*, *censorship resistance* and *deniability*.

One of the main focuses of security overlays is to provide anonymity for users, i.e., hiding the identity of participants such that they are unrecognizable amongst the community. One motivation stems from the users' desire to conceal their real network location (IP addresses). Among other reasons are the users' wishes to remain anonymous when they download or upload contents from the servers. From a technical perspective, the anonymity is accomplished through the use of encryption and tunnelling through the network, assisted with proxies that re-address the forwarded packets. Take the case of onion routing for example. Figure 2.4 depicts an onion-routed overlay network, Tor, which enables a source to encrypt his communication and tunnel traffic through multiple relaying nodes in the overly until finally communication is established with a end host on the Internet. In general, onion-routing networks are generic transport or network layer overlays capable of providing anonymity to any application. On the same while, an anonymous content and retrieval overlay network operates in a slightly different way, where interested readers can refer to Freenet [34].

The second class of functionality that security overlays provide is censorship resistance, which is designed to resist the attempts to remove, or make inaccessible, certain types of content on the Internet. The architecture for censorship resistance can be achieved in many different ways, the above stated anonymity mechanisms is one possible solution, which makes it difficult to locate the actual providers or users downloading contents. Another general strategy to avoid censorship is to automatically cache content at many different locations on the Internet. For instance, the Chord File System [58] caches file contents at distributive locations.

The final property that a typical security overlay network possesses is deniability, which is the ability to disclaim connection with or responsibility for either stored content or communications. Deniability of stored content is often accomplished by allowing nodes in a distributed overlay to store encrypted files but not the decryption keys. Each node can thus plausibly assert that they are not aware of the content of the files on their systems. Examples of this type include Publius [36] and PAST [59] systems. This property also results in censorship resistance since individual nodes are unable to choose which contents they can cache.

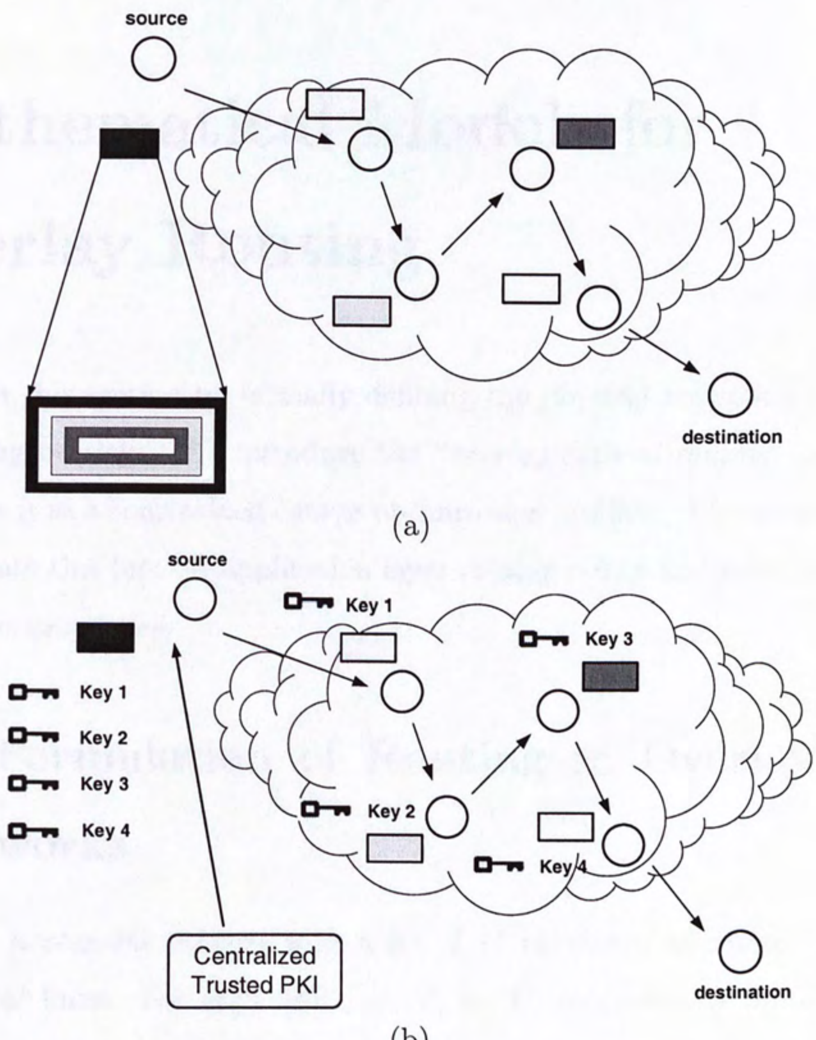


Figure 2.4: Slicing the Onion: Anonymity With PKI. (a) State of the art: onion routing with overlays, encrypt packets in layers; (b) "Slicing" the onion with PKI.

Chapter 3

Mathematical Models for Overlay Routing

We begin this section by formally defining the physical network and various co-existing overlays. We introduce the "*overlay optimal routing*" policy and formulate it as a constrained convex optimization problem. We use an example to illustrate this form of application layer routing policy and state some of its important properties.

3.1 Formulation of Routing in Overlay Networks

Consider a *physical network* with a set $\mathcal{J}$ of *resources*, which denotes a set of physical links. For each link $j \in \mathcal{J}$, let C_j represent its finite capacity (the unit is bps). The number of links in the network is finite such that $|\mathcal{J}| = m$. Let a *route* r be a non-empty subset of $\mathcal{J}$, and denote by $\mathcal{R}$ the set of all possible routes of the physical network. Since $\mathcal{J}$ is finite, the number of possible routes is also finite, say, $|\mathcal{R}| = q$. We use A to represent an $m \times q$ matrix with $A_{jr} = 1$ if $j \in r$, which indicates that link j lies on the route r, and $A_{jr} = 0$ otherwise. Thus, the matrix A defines a $0-1$ link-route indicator matrix $(A_{jr}, j \in \mathcal{J}, r \in \mathcal{R})$.

An overlay network is a connected subgraph of the underlying physical network which consists of a set of logical nodes and logical links. A logical path is interpreted as a set of logical links, each of which may consist of one or more physical links. The logical topology of the overlay network heavily depends on how this overlay is organized. With proper translation, we can map every logical path to a set of corresponding physical links. Thus, the routing matrix for overlay s can be similarly defined as $A^{(s)}$, which is a sub-matrix of A. We allow multiple overlays to co-exist on top of a physical network and these overlays may share some of the physical resources in $\mathcal{J}$. Within an overlay, there can be "*multiple*" source-sink pairs and each source-sink is associated with a traffic flow f, which has a constant traffic demand of x_f (units is bps). Though the overlay's traffic demand may be time-varying, we focus on a static underlay environment and non-time-varying overlay traffic, to capture the essential properties and implications due to overlay interactions of a less dynamic environment. For ease of presentation, we use the term *flow*, or *source-sink* interchangeably to denote a particular traffic transmission within an overlay.

Let $\mathcal{N}$ be a finite set (with $|\mathcal{N}| = n$) representing all overlays on top of a physical network. Suppose that for each overlay $s \in \mathcal{N}$, there is a finite set $\mathcal{F}_s$ of source-sink pairs. For each flow $f \in \mathcal{F}_s$, there is a set $\mathcal{R}_f$ of different paths (may not be disjoint) that can be used by flow f to deliver information from its source to its sink, where $\mathcal{R}_f$ is a non-empty subset of all possible paths $\mathcal{R}$ in the physical network. Hence, $\mathcal{R}_f$ contains all possible paths for flow f from its source to its sink. Let H be an $\sum_{s\in\mathcal{N}} |\mathcal{F}_s| \times q$ matrix and $H_{fr} = 1$ if $r \in \mathcal{R}_f$ ($f \in \mathcal{F}_s, s \in \mathcal{N}, r \in \mathcal{R}$) such that route r serves flow f in overlay s, and $H_{fr} = 0$ otherwise. Here H defines a $0-1$ indicator matrix $H = (H_{fr}, f \in \mathcal{F}_s, r \in \mathcal{R})$, specifying all possible routes that can be used by flow f in the overlay s.

An overlay can control the routing of all flows within its overlay network via an application routing policy. To achieve this, source nodes of an overlay network may choose to split and assign their traffic onto different paths so that the weighted average delay of the *whole* overlay network can be minimized. Note that this is different from the traditional overlay routing, because normally a source node in an overlay merely chooses a currently best path from a set of available paths, i.e. minimum end-to-end delay, assigning *all* its traffic along this path. Generally such flow allocation is not optimal in terms of the overlay's average performance. For each flow f in the overlay s, there is a traffic demand x_f (in terms of bps) assigned to the corresponding source-sink pair. The overlay needs to decide, for all its flows, how to assign traffic to every possible path $r \in \mathcal{R}_f$ so as to optimize its desired performance. Therefore, each flow f in the overlay s has a routing decision vector $y^{(s,f)} = \left(y_1^{(s,f)}, y_2^{(s,f)}, \ldots, y_{|\mathcal{R}_f|}^{(s,f)}\right)^{\mathcal{T}}$, where $y_k^{(s,f)}$ is the amount of traffic along path k for flow f in overlay s, and $\sum_{k=1}^{|\mathcal{R}_f|} y_k^{(s,f)} = x_f$ where $|\mathcal{R}_f|$ is the total number of paths available for the flow f. For compactness of presentation, one can rewrite the routing decision for overlay s as a concatenation of the flow vectors of all its source-sink pairs: $y^{(s)} = \left(y^{(s,f_1)}, y^{(s,f_2)}, \ldots, y^{(s,f_{|\mathcal{F}_s|})}\right)$.

We say that a flow pattern of vector $y = (y^{(s)}, s \in \mathcal{N}) = (y^{(1)}, y^{(2)}, \ldots, y^{(n)})$ *supports* traffic rate $x = (x_f, f \in \mathcal{F}_s, s \in \mathcal{N})^{\mathcal{T}}$ if $Hy = x$. In other words, summing the rate y_r on all routes r that serve flow f equals the overall traffic demand x_f. We call a flow pattern *feasible* if for $y = (y_r, r \in \mathcal{R})$, $y \geq 0$ and $Ay \leq C$, where $C = (C_j, j \in \mathcal{J})$. In other words, the aggregate rate of traffic that traverses link j is no more than the capacity C_j of link j. Table 3.1 illustrates the notation we use in defining the physical and overlay networks.

3.2 Optimal Overlay Routing Policy

Let $d_j(l_j)$ denote the delay function for physical link $j \in \mathcal{J}$, where l_j is the *aggregate* rate of traffic that traverses link j. In this work, we only assume that

$\mathcal{J}$:	a set of link resources for the physical network with $\lvert\mathcal{J}\rvert = m$.
C_j :	the finite capacity of resource $j \in \mathcal{J}$.
$\mathcal{R}$:	the set of all possible routes of the physical network with $\lvert\mathcal{R}\rvert = q$.
A_{jr}:	an indicator of whether link $j \in \mathcal{J}$ is part of route $r \in \mathcal{R}$. A is an $m \times q$ link-route indicator matrix.
$\mathcal{N}$:	a set of overlay networks with $\lvert\mathcal{N}\rvert = n$.
$\mathcal{F}_s$:	a set of flows in overlay s wherein each flow is represented by a source-sink pair and $s \in \mathcal{N}$.
$\mathcal{R}_f$:	a set of distinct paths that can be used by flow f wherein $f \in \mathcal{F}_s$.
H_{fr} :	an indicator to show whether the flow f in overlay s overlay s ($f \in \mathcal{F}_s$) uses the route $r \in \mathcal{R}$, or whether $r \in \mathcal{R}_f$. Note that H is an $\sum_{s \in \mathcal{N}} \lvert\mathcal{F}_s\rvert \times q$ matrix.
x_f :	traffic demand for flow f in the overlay s, $f \in \mathcal{F}_s$, $s \in \mathcal{N}$.
$y_k^{(s,f)}$:	amount of traffic that flow f in overlay s assigns to path k, wherein $1 \leq k \leq \lvert\mathcal{R}_f\rvert$, $f \in \mathcal{F}_s$, $s \in \mathcal{N}$ and $\sum_{k=1}^{\lvert\mathcal{R}_f\rvert} y_k^{(s,f)} = x_f$.
$y^{(s,f)}$:	the flow vector of f in overlay s, which is $\left(y_1^{(s,f)}, y_2^{(s,f)}, \ldots, y_{\lvert\mathcal{R}_f\rvert}^{(s,f)}\right)^{\mathcal{T}}$.
$y^{(s)}$:	the flow vector for all flows in overlay s, which is $\left(y^{(s,f_1)}, y^{(s,f_2)}, \ldots, y^{(s,f_{\lvert\mathcal{F}_s\rvert})}\right)$.
y :	the flow vector for the whole physical network, which is $\left(y^{(1)}, y^{(2)}, \ldots, y^{(n)}\right)$.
$A^{(s,f)}$:	an $m \times \lvert\mathcal{R}_f\rvert$ routing matrix for flow f in overlay s.
$A^{(s)}$:	the routing matrix for all flows in overlay s, which is $\left(A^{(s,f_1)}, A^{(s,f_2)}, \ldots, A^{(s,f_{\lvert\mathcal{F}_s\rvert})}\right)$.
A :	the routing matrix for all overlays in the physical network, which is $\left(A^{(1)}, A^{(2)} \ldots, A^{(n)}\right)$.

Table 3.1: Notations used to represent physical and overlay networks

the delay function is continuous, non-decreasing, and convex. Note that this is a mild assumption since it allows us to model links with a fixed propagation delay, or links delay represented by general queueing delay models. The end-to-end delay of a route is the sum of delays on all physical links that comprise this route. For a particular overlay $s \in \mathcal{N}$, the weighted average delay is

$$delay^{(s)} = \frac{1}{\sum_{f \in \mathcal{F}_s} x_f} \sum_{f \in \mathcal{F}_s} \sum_{k \in \mathcal{R}_f} y_k^{(s,f)} \cdot D_k^{(f)}, \tag{3.1}$$

where $D_k^{(f)} = \sum_{j \in k} d_j(l_j)$ is the end-to-end delay of path k for flow f. Let $L = (l_1, l_2, \ldots, l_m)^T$ denote a traffic rate vector for all physical links in the network. We define a delay function $\mathcal{D}(L) : \mathrm{R}^m \to \mathrm{R}^m$, where for each rate vector L, this function returns a delay vector for all physical links:

$$\mathcal{D}(L) = (d_1(l_1), \ldots, d_j(l_j), \ldots, d_m(l_m))^T, \ j \in \mathcal{J} \tag{3.2}$$

where $L = (l_1, l_2, ..., l_j, ..., l_m)^T, \ j \in \mathcal{J}$.

Let $A^{(s,f)}$ be an $m \times |\mathcal{R}_f|$ routing matrix for the flow f in the overlay s, of which the definition is similar to that of A. Therefore, $A^{(s,f)}$ is the partial matrix of $A^{(s)}$. For ease of presentation, we rewrite $A^{(s)} = \left(A^{(s,f_1)}, A^{(s,f_2)}, \ldots, A^{(s,f_{|\mathcal{F}_s|})}\right)$, and $A = (A^{(1)}, A^{(2)}, \ldots, A^{(n)})$. With these notations, one immediately obtains the following result of representing the traffic rate vector:

$$L = \sum_{s \in \mathcal{N}} \sum_{f \in \mathcal{F}_s} A^{(s,f)} y^{(s,f)} = \sum_{s \in \mathcal{N}} A^{(s)} y^{(s)} = Ay. \tag{3.3}$$

The weighted average delay for an overlay s can be expressed in a compact form as:

$$\begin{aligned} \mathrm{delay}^{(s)}(y^{(s)}; y^{(-s)}) &= \frac{1}{\sum_{f \in \mathcal{F}_s} x_f} \cdot y^{(s)T} \left[A^{(s)T} \mathcal{D}(L) \right] \\ &= \frac{1}{\sum_{f \in \mathcal{F}_s} x_f} \cdot y^{(s)T} \left[A^{(s)T} \mathcal{D}\left(\sum_{i \in \mathcal{N}} A^{(i)} y^{(i)} \right) \right] \end{aligned} \tag{3.4}$$

where y is a feasible flow pattern of vector $(y_r, r \in \mathcal{R}) = (y^{(1)}, \ldots, y^{(s)}, \ldots, y^{(n)})$ and $y^{(-s)}$ denotes the vector of traffic flows in other overlays except the overlay

s. Note that x_f is a fixed traffic demand, which can be treated as a constant and will not affect the optimization procedure that we will carry out in later sub-section. Therefore, the first factor in Eq.(3.4) is generally not considered when one obtain the optimal routes.

Finally, the problem of obtaining the optimal overlay routes for overlay s can be expressed as a constrained optimization problem that represents the interaction with other overlays in the network. Mathematically, we have:

$$\textbf{OVERLAY}^{(s)} \quad \Big(y^{(s)}; A, H, C, x, y^{(-s)} \Big) : \tag{3.5}$$

$$\text{Minimize} \quad y^{(s)T} \Big[A^{(s)T} \mathcal{D}(\sum_i A^{(i)} y^{(i)}) \Big] \tag{3.6}$$

$$\text{s. t.} \quad \text{for } \forall f \in \mathcal{F}_s, \ \sum_{k=1}^{|\mathcal{R}_f|} y_k^{(s,f)} = x_f,$$

$$Ay \leq C, y^{(s)} \geq 0. \tag{3.7}$$

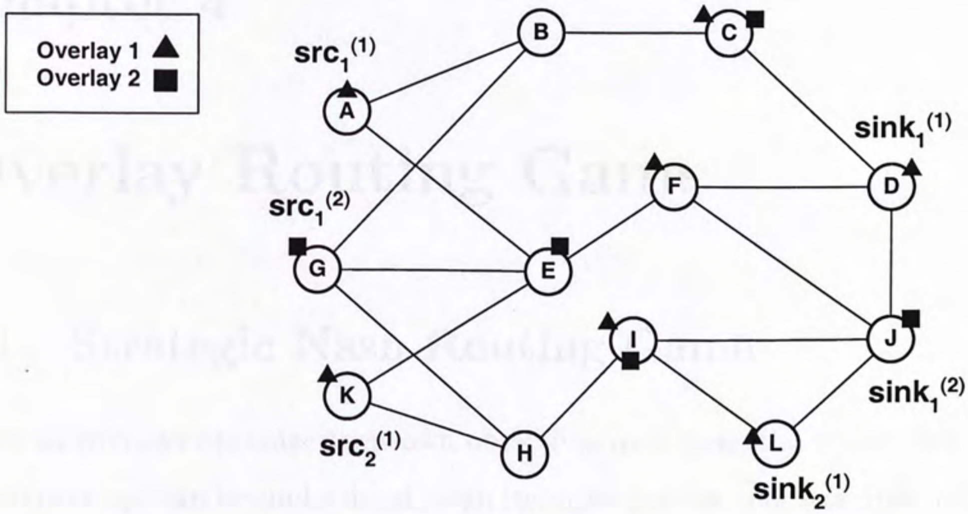
For this optimization problem, overlay s considers other overlays' routing decisions $y^{(-s)}$ as *fixed* when it makes the routing decision $y^{(s)}$ by solving the above optimization problem. Note that the objective function of this optimization problem is continuous, differentiable and convex. Since the feasible region defined by constraints in the optimization problem of Eq. (3.5) is convex and compact, the optimal value and the minimizer can be found by the Lagrangian method. Furthermore, an alternative solution is to apply the marginal cost flow approach [11] discussed shortly.

3.3 Illustration of Overlay Routing Policy

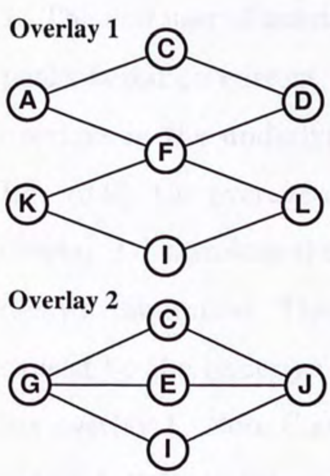
To illustrate the concept and various notations, let us consider the physical (or underlying) network depicted in Figure 3.1(a). The physical network consists of 12 physical nodes and a set of physical links. There are two co-existing overlay networks above the physical network, in which overlay 1 has two source-sinks while overlay 2 has one single source-sink pair. Each overlay is a logical network consisting of a set of physical nodes and logical links. In general, the topology of an overlay network depends on how the overlay is organized and Figure 3.1(b) shows one possibility of the logical topologies. Note that a physical node can also belong to multiple overlays, i.e., node C, I belong to both overlays. Every logical link is interpreted as a physical route between two neighbor overlay nodes, which is determined by the underlying IP level routing, i.e., the shortest path routing algorithm. The logical link to physical path mappings are depicted in Figure 3.1(b).

For each source node in both overlays, there is a set of available logical paths to the corresponding sink. Tables in Figure 3.1(b) list the set of available paths for each source-sink pair in each overlay. The routing decision vector $y^{(s)}$, $s = 1, 2$, assigns a nonnegative amount of traffic to each of these available paths such that the sum of the amount of flow over all paths is equal to the traffic demand of the flow. In this example, the traffic demand of *each* flow is 1 unit. Later, we will show in Chapter 5 the traffic rate assignment for these two overlays under three different routing policies: user selfish routing, overlay optimal routing and global optimal routing.

Note that these overlays interact with each other due to the fact that the logical paths in different overlays have overlapping physical links. For example, the logical path AF-FD, KF-FL in overlay 1 and GE-EJ in overlay 2 has the common physical link EF. Similar situation occurs for link BC, CD, HI, ... etc. As we will discuss later, since one overlay's routing decision depends on the routing decision of other overlays, there will be inevitable interaction between the routing behaviors of different overlays. An interesting question is whether this interaction will create network routing instability. We address this issue in the following section.



(a)



Overlay 1	Route	Logical path	Traffic rate
source1	$r_1^{(1,1)}$	AC-CD	$y_1^{(1,1)}$
	$r_2^{(1,1)}$	AF-FD	$1-y_1^{(1,1)}$
source2	$r_1^{(1,2)}$	KF-FL	$y_1^{(1,2)}$
	$r_2^{(1,2)}$	KI-IL	$1-y_1^{(1,2)}$

Overlay 2	Route	Logical path	Traffic rate
source1	$r_1^{(2,1)}$	GC-CJ	$y_1^{(2,1)}$
	$r_2^{(2,1)}$	GE-EJ	$y_2^{(2,1)}$
	$r_3^{(2,1)}$	GI-IJ	$1-y_1^{(2,1)}-y_2^{(2,1)}$

Logical Link	Physical Path
AC	AB-BC
CD	CD
AF	AE-EF
FD	FD
KF	KE-EF
FL	FJ-JL
KI	KH-HI
IL	IL
GC	GB-BC
CJ	CD-DJ
GE	GE
EJ	EF-FJ
GI	GH-HI
IJ	IJ

(b)

Figure 3.1: (a) A physical network with two co-existing overlays. (b) Two overlays and their routing decisions. Overlay 1 has two flows while overlay 2 has one flow. Each flow has one unit of traffic

Chapter 4

Overlay Routing Game

4.1 Strategic Nash Routing Game

Since all overlays optimize their own objectives over time, the interaction between overlays can be understood as an iterative process. For example, in the first round, there is some traffic transmission for the source-sink pair in overlay 1. The end user of overlay 1 determines the routing among all available logical paths based on current network condition, which consists of traffic from other overlays in the underlying network. By optimizing the objective function in Eq. (3.6), the average delay in overlay 1 is minimized. In the second round, overlay 2 determines the routing among its available paths so that the average delay is minimized. The routing decision is based on current network condition caused by the underlying traffic, as well as traffic from other overlays, including overlay 1. Note that although for overlay 2, the current flow allocation is optimal, it is no longer optimal for overlay 1 because of the routing decision made by overlay 2. Thus, overlay 1 needs to adaptively re-assign traffic to ensure the optimality of its performance. After some time, overlay 2 observes a performance degradation and re-assigns its traffic accordingly. This illustrates the interaction among the two overlays. In general, the interaction can be quite involved since there can be many overlays on top of a physical network. At the same time, traffic flows may begin or terminate at any time. Therefore, it is interesting to find out whether this form of interaction will converge to a stable point, and if it converges, what is the rate of convergence.

Figure 4.1 illustrates how co-existing overlays interact with each other. Each overlay periodically probes all possible logical paths within its overlay, and reallocates traffic among all paths to optimize its performance. We call the traffic allocation which optimizes the average delay in overlay s the *optimizer* for overlay s. At a particular time, each overlay optimizer consisting of flows on its logical paths since last routing update, as well as flows of the underlying traffic on physical links, constitute the input for the routing optimization problem for an overlay at the next round of routing update.

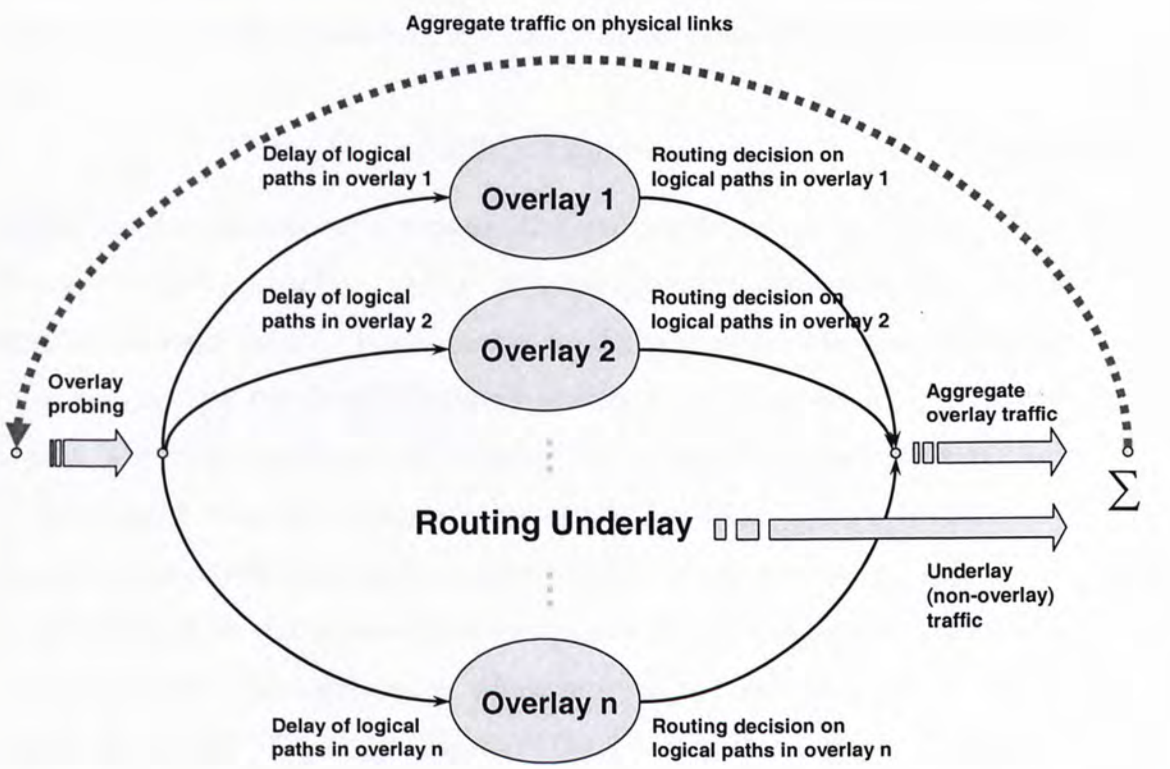


Figure 4.1: Interaction between co-existing overlays performing overlay optimal routing simultaneously.

Since multiple overlays co-exist on the same physical network, they can share some common physical links and/or nodes. Therefore, the optimization process of different overlays is *not* isolated. Namely, a change of routes by one overlay will stimulate routing updates in other overlays, which turns back to affect the routing decision of the first overlay again. In here, we show that even when individual overlay performs overlay optimal routing, the network can reach a stable equilibrium point (be immune from routing instability).

Based on the mathematical formulation in Chapter 3, one can model the interaction between multiple co-existing overlays as a Nash routing game. First, a finite set of players $\mathcal{N}$ consists of all overlays, where $\mathcal{N} = \{1, 2, \ldots, n\}$. Second, the strategy that an overlay can take is a vector of flow among all available paths, where each component of the vector is nonnegative and satisfies the transmission demand. Further, the rate vector should satisfy the capacity constraint of each link. Formally, the set of action profiles for overlay s is:

$$\Gamma_s = \{y^{(s)} \mid y^{(s)} \in \mathbb{R}_+^{r_s},\ (Hy)_s = x_s,\ Ay \leq C\} \tag{4.1}$$

where $\mathbb{R}_+$ denotes the set of nonnegative real numbers and $r_s = \sum_{f \in \mathcal{F}_S} |\mathcal{R}_f|$. If $y = (y^{(1)}, y^{(2)}, \ldots, y^{(s)}, \ldots, y^{(n)})^T$ satisfies the above constraint, then it is a feasible strategy profile. Furthermore, we define a preference relation $\succsim_s$ for player s. For any two feasible strategy profiles y and y', we say $y \succsim_s y'$ (player s prefers strategy profile y to y') if $\text{delay}^{(s)}(y) \leq \text{delay}^{(s)}(y')$, where $\text{delay}^{(s)}(y)$ is the average delay for overlay s as defined by Eq. (3.1). Thus it is equivalent to define the payoff function for player s as the negative of $\text{delay}^{(s)}(y)$.

We formulate the interaction between co-existing overlays as an n-player non-cooperative strategic game, which we call the *overlay optimal routing game*: $G_{\text{overlay}}\langle \mathcal{N}, (\Gamma_s), (\succsim_s) \rangle$. We have the following definition of *Nash Equilibrium* for this overlay routing game G_{overlay}.

Definition 4.1 A feasible strategy profile $y^* \in \Gamma_1 \times \ldots \times \Gamma_n$, $y^* = (y^{*(1)}, \ldots, y^{*(s)}, \ldots, y^{*(n)})^T$ is called a Nash equilibrium if for *every* player $s \in \mathcal{N}$, $\text{delay}^{(s)}(y^{*(1)}, \ldots, y^{*(s)}, \ldots, y^{*(n)})$ is less than or equal to

$\text{delay}^{(s)}(y^{*(1)}, \ldots, y'^{(s)}, \ldots, y^{*(n)})$ for any other feasible strategy profile $y'^{(s)}$.

The analysis of this strategic routing game is based on a discrete time model, that is, each overlay calculates its optimal strategy at its own update frequencies. We assume one overlay has sufficient time to complete this optimization before other overlays begin their optimization processes.

4.2 Stable Property of Overlay Optimal Routing

We now show that the overlay routing game $G_{\text{overlay}}\langle \mathcal{N}, (\Gamma_s), (\succsim_s)\rangle$ possesses a Nash equilibrium for a general network setting as long as the delay function for each physical link is continuous, non-decreasing and strictly convex.

Lemma 4.2 A strategic game $\langle \mathcal{N}, (\Gamma_s), (\succsim_s)\rangle$ has a Nash equilibrium if for all players $s \in \mathcal{N}$ that (1) the set Γ_s of action profiles for player s is a nonempty compact convex subset of a Euclidean space and, (2) the preference relation $\succsim_s$ is (a) continuous and (b) quasi-concave on Γ_s.

Proof: Note that Lemma 4.2 provides a standard approach to prove the existence of Nash equilibrium in a strategic game. For detailed proof of lemma 4.2, readers can refer to [60]. ∎

Lemma 4.3 In the overlay optimal routing game $G_{\text{overlay}}\langle \mathcal{N}, (\Gamma_s), (\succsim_s)\rangle$, the set Γ_s of action profiles for overlay s is a nonempty compact convex subset of a Euclidean space.

Proof: The set of action profiles for overlay s is defined as $\Gamma_s = \{y^{(s)} \mid y^{(s)} \in \mathbb{R}_+^{r_s},\ (Hy)_s = x_s,\ Ay \leq C\}$. Since the feasible region is closed and bounded, Γ_s is compact, and because all constraints are affine functions, the feasible region which is the intersection of half-spaces and hyperplanes, is also convex. ∎

Lemma 4.4 The preference relation $\succsim_s$ in an overlay optimal routing game $G_{\text{overlay}}\langle\mathcal{N}, (\Gamma_s), (\succsim_s)\rangle$ is continuous and quasi-concave on Γ_s.

Proof: Please refer to the appendix A. ∎

Theorem 4.5 In the optimal overlay routing game $G_{\text{overlay}}\langle\mathcal{N}, (\Gamma_s), (\succsim_s)\rangle$, there exists a Nash Equilibrium if the delay function $\text{delay}^{(s)}(y^{(s)}; y^{(-s)})$ is continuous, non-decreasing and strictly convex.

Proof: By lemma 4.2, 4.3 and 4.4, we can immediately show this result. Some remarks: the proof of the theorem follows a standard approach [60] which proves the existence of Nash equilibrium in a non-cooperative strategic game. The proof can also be obtained by applying Rosen's Theorem [61]. ∎

4.3 Routing Game in Other Forms

Here we introduce some other models of overlay routing and interpret the routing mechanisms studied in previous papers as a routing game.

For the traditional selfish routing studied in [62], each user controls a non-negligible amount of traffic and does not split traffic among all available paths. Each user always selects a single route with the minimum end-to-end delay from the set of possible paths. This can be formulated as a pure strategy routing game. In such a routing game, there may not exist a Nash equilibrium so this is the justification why oscillatory behavior exists in selfish routing. On the other hand, if each user uses a mixed strategy and selects every possible path probabilistically so that the expected end-to-end delay is minimized, there always exists an equilibrium and it is addressed in [63]. The existence and convergence of this routing mechanism can be explained by the fact that every finite strategic game has a mixed strategy Nash equilibrium [60, 64].

For the overlay optimal routing game we consider in this paper, each overlay has the same information about the routing game and they do not know *how* the routing decisions of other overlays are being made. Now consider another type of game called the Stackelberg game. For this type of game, some overlay has a prior knowledge about the routing decisions of other overlays. Given this knowledge, the overlay will have the *first-move advantage*. In this case, we consider a particular routing strategy $y^{(s)}$ by overlay s. Overlay s is assumed to have the ability to calculate other overlays' best response according to $y^{(s)}$. Then overlay s optimizes over all actions to get the optimal solution so that it achieves the minimum average delay. For instance, for a simple network with only two overlays, overlay 1 has a first-move advantage is actually solving the following problem:

$$\begin{aligned} \textbf{Stackelberg} \quad & \textbf{Overlay Routing}^{(1)}(\ y^{(1)}\ ): \\ \text{minimize} \quad & \text{delay}^{(1)}(y^{(1)}, y^{(2)*}) \\ \text{s.t.} \quad & Hy = x, Ay \leq C \\ & y^{(1)} \geq 0 \\ \text{where} \quad & y^{(2)*} = \text{argmin delay}^{(2)}(y^{(2)}; y^{(1)}) \end{aligned}$$

This type of game models the scenario in which some overlays have more information than other overlays and can gain by doing an offline computation. Similar topics were discussed in [48] in which a Stackelberg game occurs between an overlay routing decision and an ISP traffic engineering decision.

Chapter 5

Comparison of Routing Strategies: A Spectrum of Efficiency

In this section, we present the relationship between global optimal routing [65], selfish routing and overlay optimal routing, and show that these routing schemes are implicitly solving three different types of system optimization problems. However, there is a fundamental distinction between their objectives and this provides users a spectrum of efficiency and fairness for operating an overlay network. Let us use the same network illustrated in Figure 3.1 to show how these three routing strategies work. Note that the delay of each link is a function of the aggregate rate of traffic that traverses this link. In Table 5.1, we list the delay functions of the physical links. We denote by $d_j(l_j)$ the delay function of link j and l_j as the aggregate traffic rate of link j. For simplicity of presentation, the first set of links in Table 5.1 has a "*linear*" delay function (i.e., delay proportional to the aggregate traffic), while the rest of the links in the network have a constant delay of zero or one unit of time.

Let us consider the overlay routing decision vector: $y^{(1,1)}=(y_1^{(1,1)},1-y_1^{(1,1)})^{\mathcal{T}}$ denotes the traffic rate through routes *AC*-*CD*, *AF*-*FD* for the first flow in overlay 1 respectively, and $y^{(1,2)}= (y_1^{(1,2)},1-y_1^{(1,2)})^{\mathcal{T}}$ denotes traffic rate through

delay function	physical links
$d_j(l_j) = l_j$	AB, BC, EF, FJ, HI
$d_j(l_j) = 0$	FD, DJ, KE, GE, JL
$d_j(l_j) = 1$	$AE, GB, GH, KH, CD, IJ, IL$

Table 5.1: Delay functions of physical links

routes KF-FL, KI-IL for the second flow in overlay 1 respectively. Similarly, $y^{(2,1)} = (y_1^{(2,1)}, y_2^{(2,1)}, 1 - y_1^{(2,1)} - y_2^{(2,1)})$ denotes the traffic rate through routes GC-CJ, GE-EJ and GI-IJ for the flow in overlay 2. We assume the traffic demands for all the flows in both overlays is one unit, and there is no underlying traffic in the physical network. For simplicity of presentation, we assume the capacity for each link is sufficiently large to support the given traffic flow demand.

5.1 Global Optimal Routing

In this section, we provide a first attempt to explore the operating point to achieve a global optimum. Note that an overlay exists on top of a physical network, which is managed by an ISP. Usually, an ISP may have a different objective in mind, for example, an ISP may want to perform traffic engineering so as to reduce a network bottleneck on a physical link, or an ISP may want to control the physical network so that the overall network performance can be guaranteed. One way to perform traffic engineering is to use the notion of *global optimal routing*.

Routing algorithms that achieves global optimality was studied by Bertsekas and Gallager [65]. Such global optimal routing achieves a minimum average latency for "*all*" the traffic in the underlay network. Global optimal routing can be formulated as a centralized optimization problem. To illustrate, let us use the same network depicted in Figure 3.1 to show how this routing strategy work.

As mentioned above, the objective of the global optimal routing is to minimize the average latency for *all* traffic in the network. Formally, it can be formulated as the following constrained optimization problem:

$$\textbf{GLOBAL} \quad (\,y; A, H, C, x) : \tag{5.1}$$

$$\text{minimize} \quad y^{\mathcal{T}}\left[A^{\mathcal{T}}\mathcal{D}(Ay)\right] \tag{5.2}$$

$$\text{s. t.} \quad Hy = x, Ay \leq C \tag{5.3}$$

$$\text{over} \quad y \geq 0 \tag{5.4}$$

The Karush-Kuhn-Tucker (KKT) conditions [11] for solving this form of optimization problem is that all routes with non-zero traffic serving the same source-sink pair must have the same end-to-end "*length*", where the length of a link is interpreted as the first derivative of traffic rate times delay at current traffic level, which is $(l_j \cdot d_j(l_j))'$. Furthermore, the length of these routes is the minimum among all available routes serving the source-sink pair. We write the conditions for source-sink pair in overlay 1 as follows:

$$D_1^{(1,1)} = 2y_1^{(1,1)} + 2(y_1^{(1,1)} + y_1^{(2,1)}) + 1 \qquad \begin{cases} = u_1 & \text{if} \;\; y_1^{(1,1)} > 0 \\ \geq u_1 & \text{if} \;\; y_1^{(1,1)} = 0 \end{cases}$$

$$D_2^{(1,1)} = 1 + 2(y_2^{(2,1)} + 1 - y_1^{(1,1)} + y_1^{(1,2)}) \qquad \begin{cases} = u_1 & \text{if} \;\; 1 - y_1^{(1,1)} > 0 \\ \geq u_1 & \text{if} \;\; 1 - y_1^{(1,1)} = 0 \end{cases}$$

$$D_1^{(1,2)} = 2(1 - y_1^{(1,1)} + y_2^{(2,1)} + y_1^{(1,2)}) + 2(y_2^{(2,1)} + y_1^{(1,2)}) \qquad \begin{cases} = u_2 & \text{if} \;\; y_1^{(1,2)} > 0 \\ \geq u_2 & \text{if} \;\; y_1^{(1,2)} = 0 \end{cases}$$

$$D_2^{(1,2)} = 1 + 2(1 - y_1^{(2,1)} - y_2^{(2,1)} + 1 - y_1^{(1,2)}) + 1 \qquad \begin{cases} = u_2 & \text{if} \;\; 1 - y_1^{(1,2)} > 0 \\ \geq u_2 & \text{if} \;\; 1 - y_1^{(1,2)} = 0 \end{cases}$$

For the source-sink pair of overlay 2, the KTT conditions are:

$$D_1^{(2,1)} = 1 + 2(y_1^{(1,1)} + y_1^{(2,1)}) \quad \begin{cases} = u_3 & \text{if } y_1^{(2,1)} > 0 \\ \geq u_3 & \text{if } y_1^{(2,1)} = 0 \end{cases}$$

$$D_2^{(2,1)} = 2(y_2^{(2,1)} + 1 - y_1^{(1,1)} + y_1^{(1,2)}) + 2(y_2^{(2,1)} + y_1^{(1,2)}) \quad \begin{cases} = u_3 & \text{if } y_2^{(2,1)} > 0 \\ \geq u_3 & \text{if } y_2^{(2,1)} = 0 \end{cases}$$

$$D_3^{(2,1)} = 1 + 2(1 - y_1^{(2,1)} - y_2^{(2,1)} + 1 - y_1^{(1,2)}) + 1 \quad \begin{cases} = u_3 & \text{if } 1 - y_1^{(2,1)} - y_2^{(2,1)} > 0 \\ \geq u_3 & \text{if } 1 - y_1^{(2,1)} - y_2^{(2,1)} = 0 \end{cases}$$

where u_1, u_2 and u_3 refer to the minimum end-to-end first derivative length for three flows in the two overlays respectively (also, they are the Lagrange multipliers for this optimization problem). Solving these two sets of equations provide the optimal solution of $y^{(1,1)} = (\frac{3}{8}, \frac{5}{8})$, $y^{(1,2)} = (\frac{1}{8}, \frac{7}{8})$ and $y^{(2,1)} = (\frac{1}{2}, \frac{1}{2}, 0)$. Note that, although the solution is unique in this example, in general, the solution for a global optimal routing problem may not be unique.

Classical global optimal routing [65] is a centralized routing policy and the objective of it is to minimize the weighted average delay for "*all*" traffic in the physical network. Due to lack of space, we will not repeat its theoretical formulation here (readers can refer to [65]) but just give the optimal solution for the example illustrated in Figure 3.1: $y^{(1,1)} = (\frac{3}{8}, \frac{5}{8}), y^{(1,2)} = (\frac{1}{8}, \frac{7}{8})$ and $y^{(2,1)} = (\frac{1}{2}, \frac{1}{2}, 0)$. Note that in general, the solution to the global optimal routing may not be unique.

5.2 Selfish User Routing

Selfish routing, on the other hand, is a greedy strategy which maximizes local benefit for a single user. Theoretically, each user controls an *infinitesimally small portion* of traffic and selects a path with the shortest delay. In real networks, the smallest unit is a packet. So the closest approximation of implementing the selfish routing is to assign each packet to the path which currently has the shortest delay. To allow end users to choose routes by themselves, either source routing (e.g. Nimrod [66]) or overlay routing (e.g. Detour [9] or RON [2]) can be used. In overlay routing, end users can find multiple overlay paths, with the assistance of other nodes in the same overlay relaying packets to the sink. Based on the result in [11], this is equivalent to solving the following constrained convex optimization problem:

$$\begin{array}{rl} \textbf{SELFISH} & (\,y; A, H, C, x\,): \\ \text{Minimize} & \displaystyle\sum_{j\in\mathcal{J}} \int_0^{l_j} d_j(t)dt, \\ \text{s. t.} & Hy = x;\ Ay \le C;\ L = Ay;\ y \ge 0. \end{array}$$

The Karush-Kuhn-Tucker (KKT) condition for the solution of selfish routing is that every route with non-zero traffic serving the same source-sink pair has to have the same end-to-end delay, moreover, it should be the minimum among all available routes. Thus, one can express the KKT conditions formally as follows. Let $D_k^{(s,f)}$ denote the end-to-end delay of the k-th path for flow f in overlay s. For overlay 1 we have:

$$D_1^{(1,1)} = y_1^{(1,1)} + (y_1^{(1,1)} + y_1^{(2,1)}) + 1 \quad \begin{cases} = u_4 & \text{if } \ y_1^{(1,1)} > 0 \\ \ge u_4 & \text{if } \ y_1^{(1,1)} = 0 \end{cases}$$

$$D_2^{(1,1)} = 1 + (y_2^{(2,1)} + 1 - y_1^{(1,1)} + y_1^{(1,2)}) \quad \begin{cases} = u_4 & \text{if } \ 1 - y_1^{(1,1)} > 0 \\ \ge u_4 & \text{if } \ 1 - y_1^{(1,1)} = 0 \end{cases}$$

$$D_1^{(1,2)} = (1 - y_1^{(1,1)} + y_2^{(2,1)} + y_1^{(1,2)}) + (y_2^{(2,1)} + y_1^{(1,2)}) \quad \begin{cases} = u_5 & \text{if } \ y_1^{(1,2)} > 0 \\ \ge u_5 & \text{if } \ y_1^{(1,2)} = 0 \end{cases}$$

$$D_2^{(1,2)} = 1 + (1 - y_1^{(2,1)} - y_2^{(2,1)} + 1 - y_1^{(1,2)}) + 1 \quad \begin{cases} = u_5 & \text{if } \ 1 - y_1^{(1,2)} > 0 \\ \ge u_5 & \text{if } \ 1 - y_1^{(1,2)} = 0 \end{cases}$$

For overlay 2, the KKT conditions are:

$$D_1^{(2,1)} = 1 + (y_1^{(1,1)} + y_1^{(2,1)}) + 1 \quad \begin{cases} = u_6 & \text{if } \; y_1^{(2,1)} > 0 \\ \geq u_6 & \text{if } \; y_1^{(2,1)} = 0 \end{cases}$$

$$D_2^{(2,1)} = (y_2^{(2,1)} + 1 - y_1^{(1,1)} + y_1^{(1,2)}) + (y_2^{(2,1)} + y_1^{(1,2)}) \quad \begin{cases} = u_6 & \text{if } \; y_2^{(2,1)} > 0 \\ \geq u_6 & \text{if } \; y_2^{(2,1)} = 0 \end{cases}$$

$$D_3^{(2,1)} = 1 + (1 - y_1^{(2,1)} - y_2^{(2,1)} + 1 - y_1^{(1,2)}) + 1 \quad \begin{cases} = u_6 & \text{if } \; 1 - y_1^{(2,1)} - y_2^{(2,1)} > 0 \\ \geq u_6 & \text{if } \; 1 - y_1^{(2,1)} - y_2^{(2,1)} = 0 \end{cases}$$

Here, u_7, u_8 and u_9 refer to the minimum end-to-end delays for three flows in the two overlays respectively. Solving these equations yields $y^{(1,1)} = (\frac{3}{4}, \frac{1}{4}), y^{(1,2)} = (\frac{3}{4}, \frac{1}{4})$ and $y^{(2,1)} = (0, \frac{1}{2}, \frac{1}{2})$. Also note that this solution is not unique. In general, the equilibrium point for a selfish routing scheme may not be unique. Another important point to note is that in a realistic network environment, it is impractical for end users to select the shortest delay path on a per-packet basis because this requires a significant amount of overhead in probing and estimating the end-to-end delay. To reduce the overhead and at the same time obtain reasonable performance, one may attempt to do routing based on a flow by flow basis and perform the traffic assignment on a set of possible paths such that whenever a flow is generated, it is routed along the path with current minimum end-to-end delay. The routing decision is fixed until the next routing update, in which the whole flow is routed along the new shortest delay path. Unfortunately, this type of selfish routing will cause severe routing oscillations [45] which leads to highly varied link utilization due to simultaneous routing update.

5.3 Optimal Overlay Routing

While selfish routing represents egoism from an end users' perspective, overlay optimal routing represents such selfishness from a larger entity's point of view. In the special case of two overlays, the interaction can be modeled similarly as a duopoly game [64]. For an individual overlay, overlay optimal routing achieves optimum within one overlay's range, like the way global optimal routing achieves global optimality within the *entire* underlying physical network through traffic engineering. Overlays can obtain, from a common routing underlay, the routing information of the underlying network, *e.g.*, topology of physical network, delay function of physical links, current traffic rate of different links [12]. With this information, overlays adaptively regulate their traffic so that the average delay of the overall traffic is minimized. In the case that there are multiple overlays co-existing above the same physical network, it is very likely that there will be partially overlapping paths. Moreover, one physical node may belong to several overlays. Thus, there will be interaction if multiple overlays optimize their performance simultaneously.

To illustrate the routing decision and the existence of the Nash equilibrium of the overlay routing game, let us use the same example illustrated in Figure 3.1. For the overlay optimal routing, the equilibrium traffic rates for both overlays can be determined as follows. We calculate for each overlay their best response to the routing decision of the other overlay. Given that the routing strategy of other overlays stay fixed during the period of the optimization process, it is the solution of the convex optimization problem in Eq (3.5). That is, for every path with non-zero traffic serving the same source-sink pair, the "*length*" of these paths should be the same. In here, the length of an edge is defined as the first derivative of the weighted delay on this edge within its own overlay. Namely, one can rewrite the first derivative of the weighted delay for each link j as $(l_j \cdot d_j(l_j + l_j^*))' = d_j(l_j + l_j^*) + l_j \cdot d_j'(l_j + l_j^*)$, where l_j is the aggregate rate of traffic belonging to its own overlay that traverses link j, and l_j^* denotes the aggregate traffic from other overlays. Furthermore, the length of the selected paths should be the minimum among all available routes serving this source-sink pair.

Let $D_k^{\prime(s,f)} = \sum_{j\in k}(l_j \cdot d_j(l_j + l_j^*))'$ be the first derivative length of the k-th path for flow f in overlay s, wherein l_j is the rate of traffic in overlay s that traverses link j. First, given overlay two's routing strategy as $y^{(2,1)} = (y_1^{(2,1)}, y_2^{(2,1)}, 1-y_1^{(2,1)}-y_2^{(2,2)})$, the best response for overlay one $y^{*(1,1)} = (y_1^{*(1,1)}, 1-y_1^{*(1,1)})$ and $y^{*(1,2)} = (y_1^{*(1,2)}, 1-y_1^{*(1,2)})$ should satisfy the following KKT conditions:

$$D_1^{\prime(1,1)} = 2y_1^{*(1,1)} + 2y_1^{*(1,1)} + y_1^{(2,1)} + 1 \quad \begin{cases} = u_7 & \text{if } \ y_1^{*(1,1)} > 0 \\ \geq u_7 & \text{if } \ y_1^{*(1,1)} = 0 \end{cases}$$

$$D_2^{\prime(1,1)} = 1 + 2(1 - y_1^{*(1,1)} + y_1^{*(1,2)}) + y_2^{(2,1)} \quad \begin{cases} = u_7 & \text{if } \ 1 - y_1^{*(1,1)} > 0 \\ \geq u_7 & \text{if } \ 1 - y_1^{*(1,1)} = 0 \end{cases}$$

$$D_1^{\prime(1,2)} = 2(1 - y_1^{*(1,1)} + y_1^{*(1,2)}) + y_2^{(2,1)} + 2y_1^{*(1,2)} + y_2^{(2,1)} \quad \begin{cases} = u_8 & \text{if } \ y_1^{*(1,2)} > 0 \\ \geq u_8 & \text{if } \ y_1^{*(1,2)} = 0 \end{cases}$$

$$D_2^{\prime(1,2)} = 1 + 2(1 - y_1^{*(1,2)}) + 1 - y_1^{(2,1)} - y_2^{(2,1)} + 1 \quad \begin{cases} = u_8 & \text{if } \ 1 - y_1^{*(1,2)} > 0 \\ \geq u_8 & \text{if } \ 1 - y_1^{*(1,2)} = 0 \end{cases}$$

Similarly, given overlay one's routing strategy: $y^{(1,1)} = (y_1^{(1,1)}, 1 - y_1^{(1,1)})$ and $y^{(1,2)} = (y_1^{(1,2)}, 1-y_1^{(1,2)})$, the best response of overlay two $y^{*(2,1)} = (y_1^{*(2,1)}, y_2^{*(2,1)}, 1-y_1^{*(2,1)} - y_2^{*(2,1)})$ should satisfy the following KKT conditions:

$$D_1^{\prime(2,1)} = 1 + 2y_1^{*(2,1)} + y_1^{(1,1)} + 1 \quad \begin{cases} = u_9 & \text{if } \ y_1^{*(2,1)} > 0 \\ \geq u_9 & \text{if } \ y_1^{*(2,1)} = 0 \end{cases}$$

$$D_2^{\prime(2,1)} = 2y_2^{*(2,1)} + y_1^{(1,2)} + 1 - y_1^{(1,1)} + 2y_2^{*(2,1)} + y_1^{(1,2)} \quad \begin{cases} = u_9 & \text{if } \ y_2^{*(2,1)} > 0 \\ \geq u_9 & \text{if } \ y_2^{*(2,1)} = 0 \end{cases}$$

$$D_3^{\prime(2,1)} = 1 + 2(1 - y_1^{*(2,1)} - y_2^{*(2,1)}) + 1 - y_1^{(1,2)} + 1 \quad \begin{cases} = u_9 & \text{if } \ 1 - y_1^{*(2,1)} - y_2^{*(2,1)} > 0 \\ \geq u_9 & \text{if } \ 1 - y_1^{*(2,1)} - y_2^{*(2,1)} = 0 \end{cases}$$

Here u_4, u_5 and u_6 refer to the minimum end-to-end first derivative length for the three flows as discussed above, and they are the Lagrange multipliers for the corresponding optimization problem. In the Nash equilibrium, each overlay's routing strategy should be the best response to the other overlay's. Therefore, the strategy profile $(y^{*(1,1)}, y^{*(1,2)}, y^{*(2,1)})$ in the Nash equilibrium should satisfy all KKT conditions listed above. Solving the set of equations, we have the results that $y^{*(1,1)} = (\frac{35}{72}, \frac{37}{72}), y^{*(1,2)} = (\frac{83}{216}, \frac{133}{216})$ and $y^{*(2,1)} = (\frac{11}{36}, \frac{49}{108}, \frac{13}{54})$.

5.4 Performance Comparison

Finally, Table 5.2 depicts the performance measures in terms of weighted average delay as defined in Equation (3.1) of the three routing strategies. From the

Weighted delay $delay^{(s)} = \frac{1}{\sum_{f\in\mathcal{F}_s} x_f} \sum_{f\in\mathcal{F}_s} \sum_{k\in\mathcal{R}_f} y_k^{(s,f)} \cdot D_k^{(f)}$	**Overlay One**	**Overlay Two**	**Overall delay**
Centralized global optimal routing	2.50	2.38	2.46
Selfish routing	2.63	2.75	2.67
Overlay optimal routing	2.46	2.53	2.48

Table 5.2: End-to-end delays for different routing strategies

table, one observes that these three routing strategies achieve distinctly different overall performances, as well as individual overlay performances, which represents a spectrum of efficiency and fairness for different routing strategies. In terms of the weighted delay for overall traffic, global optimal routing achieves the best while selfish routing achieves the worst. For individual overlays, selfish routing is still the worst among three of them, which confirms the inherent inefficiency discovered in previous studies [11]. When one considers the performance of the overlay optimal routing, we observe that the weighted delay of overlay 1 decreases while the weighted delay of overlay 2 increases, when compared to their respective performance achieved using the global optimal routing. It is interesting to find one overlay achieves a better performance at the expense of the other overlays. In later sections, we will explore the fairness and resource allocation issues of this phenomenon.

Chapter 6

Simulations on Routing Game

6.1 Fluid Level Simulation

In here, we illustrate the existence of the Nash Equilibrium point via simulation. We use both the fluid-based and the packet-based (NS-2) methods to carry out our simulations.

Let us consider an IP network with multiple overlays constructed on top of a physical network domain. We assume all overlays obtain the routing information from a common routing underlay [12] such that different overlays are transparent to each other. We consider a scenario with six overlay networks deployed above the same physical network of a tier-1 ISP backbone topology, as shown in Figure 6.1. In each of the six overlays, there is one source-sink pair (denoted as S and D) and a set of available paths that serve the source-sink pair. The available paths in different overlays can be fully or partially overlapping. In the simulation, we pre-assign a set of available paths for each overlay and the number of available paths within overlay $1, 2, 3, 4, 5$ and 6 is $8, 8, 8, 2, 8, 4$ respectively.

We set the traffic demands for all the source-sinks to be equal, $x_s = 1.0$Mbps, $s \in \mathcal{N}$. To model the interaction process in real networks, each overlay assumes the delay function for each physical link is derived from an M/M/1 queueing model. We set the capacity for those central links as 1.5Mbps, and

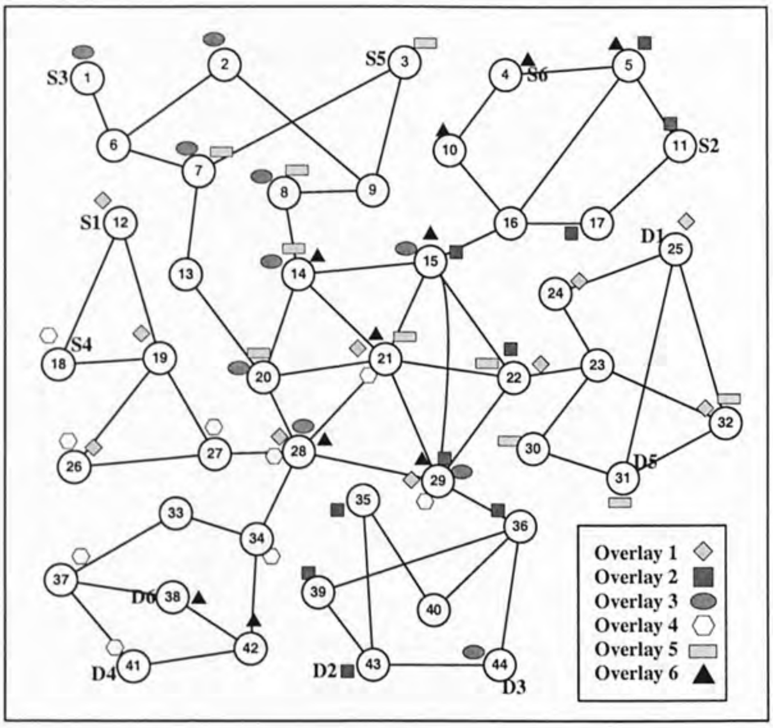


Figure 6.1: Fluid simulation: Topology of multiple co-existing overlay networks

10Mbps for other links. Therefore, we simulate a congested network such that overlays compete for the limited common resources (bandwidth). During each iteration of the routing optimization, overlays decide how to split the given traffic demand among all possible paths such that the weighted average delay is minimized. For each overlay, we set the time interval between two consecutive routing updates as a real number evenly distributed over $(0, 1)$. The six overlays start their transmission at $\tilde{t}$, which is uniformly distributed between $(0, 1.5]$.

Figure 6.2 illustrates the average delay for each of the six overlays versus simulation time. Some observations can be made. First of all, note that as more overlays start their traffic transmission, they respond quickly to the routing decisions of other existing overlays and one can observe an increase of the average delay as the network becomes more congested. Secondly, after a few rounds of routing optimizations, the average delay for different overlays quickly converge to fixed levels. One concludes that the Nash equilibrium has been reached.

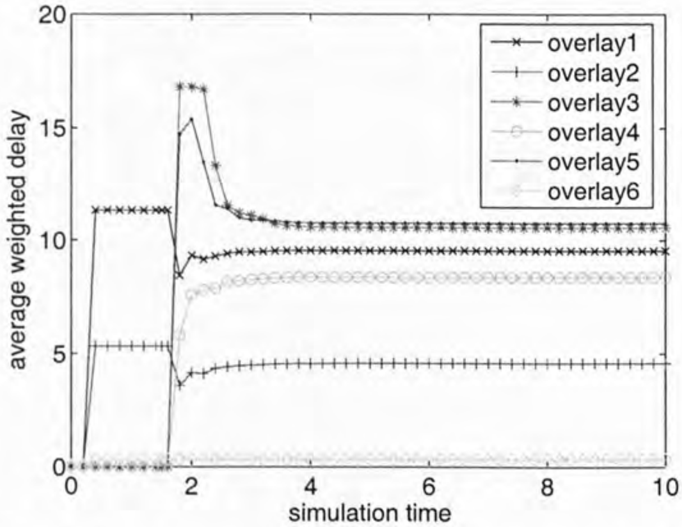


Figure 6.2: Fluid Simulation: Average delay for six overlays as a function of time

In Figure 6.3, we show how the routing decision of each overlay changes as a function of time. In each graph, each curve shows the traffic rate assigned to each of all prescribed paths for a particular overlay. Some important observations are made. First, we observe that each overlay reacts according to the routing decision of other overlays. Secondly, for each overlay, traffic assigned to a particular path gradually converges to a fixed level, compared to the convergence of the average delay in that overlay. The convergence of traffic assigned to different paths can also be viewed as a sign of reaching the Nash equilibrium. Lastly, we observe that for some overlays, the convergence rate is *relatively slow*, as compared to other overlays in the system. For example, overlay 2 takes much longer to converge than overlay 1. This occurs because some paths of overlay 2 contain congested links, and any routing decision made by other overlays significantly impacts the routing decision of overlay 2.

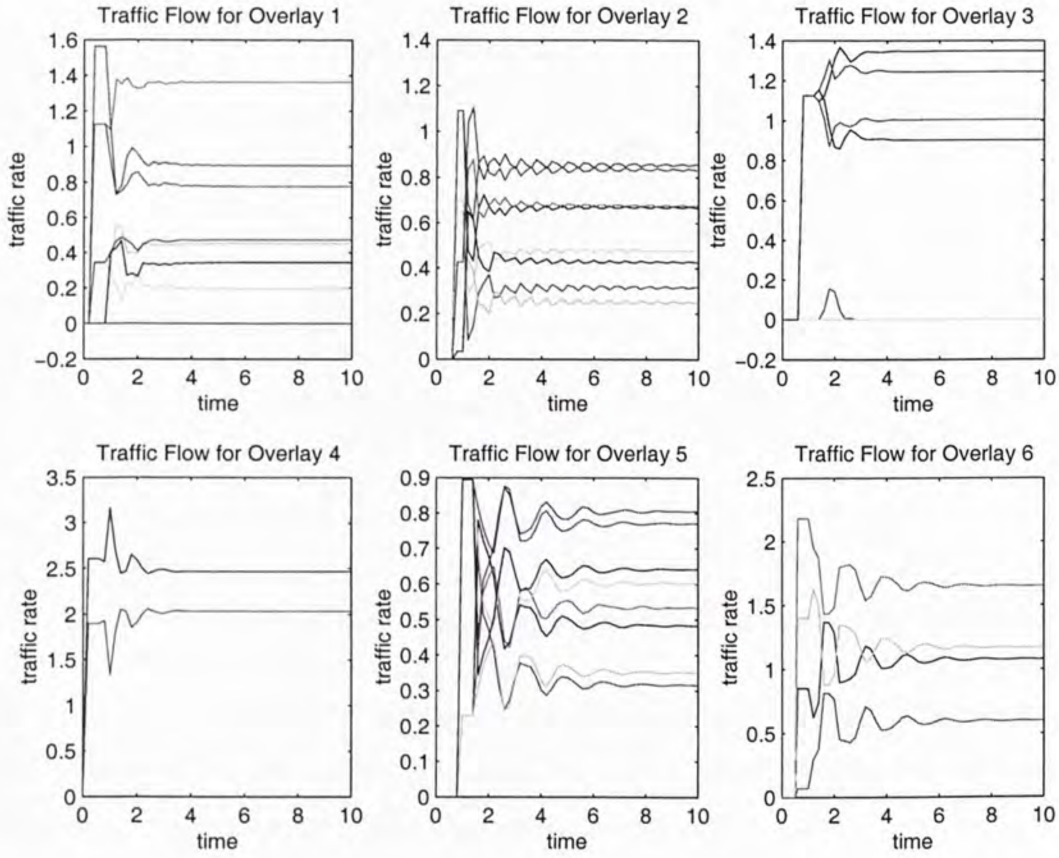


Figure 6.3: Simulation: Flow allocation for six overlays as a function of time

6.2 Packet Level Simulation

Let us consider a packet-level simulation using NS-2. The goal is to verify that the previously stated properties and some problems do exist in a realistic IP network. We allow the network to have dynamic join and leave of overlay traffic, and obtain various performance measures such as the throughput of different links. We then compare the performance of different overlays at different phases of the simulation.

Figure 6.4 depicts the physical network topology as well as the logical topologies of overlays under our packet-level simulation. There are three overlay networks co-existing on top of the physical network. We specifically simulate a moderate size network such that some physical links are highly shared by

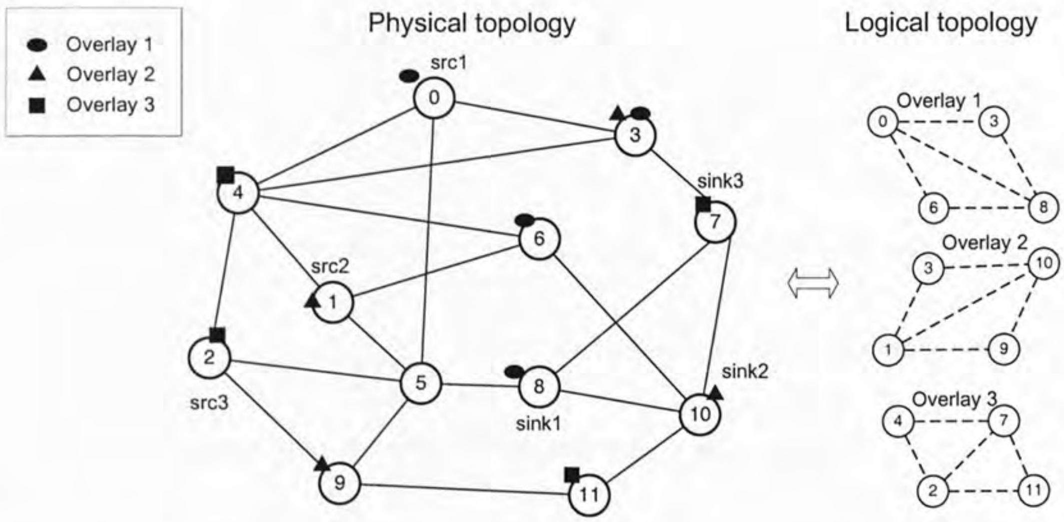


Figure 6.4: Topology of the physical network under the packet-level simulation

these three overlays. For example, link $N3-N7$ is shared by all three overlays while links $N7-N8$, $N7-N10$ and $N8-N10$ are shared by two overlays. The underlying routing mechanism uses shortest path routing and all physical links have the same capacity of 40 units (or Mbps). The traffic demand of the three overlays are $x_1 = x_2 = x_3 = 30$ units (or Mbps). Thus this simulates a scenario wherein the network is moderately congested and the shared links constitute performance bottlenecks. We posit that a shared routing underlay provides the necessary information for overlays to perform a routing optimization, such as the capacity of all links and the instantaneous throughput of these links [12]. The packet's inter-arrival time and packet size are exponentially distributed such that each link can be modelled as an $M/M/1$ queue. With these configurations, an overlay can derive the analytical expression of the delay functions of these physical links and perform an optimization based on the information it obtains from the common routing underlay.

Figure 6.5 depicts the traffic dynamics (i.e., on/off state) of these three overlays. The whole simulation lasts 2,000 seconds. At time $t = 0$, overlay 1

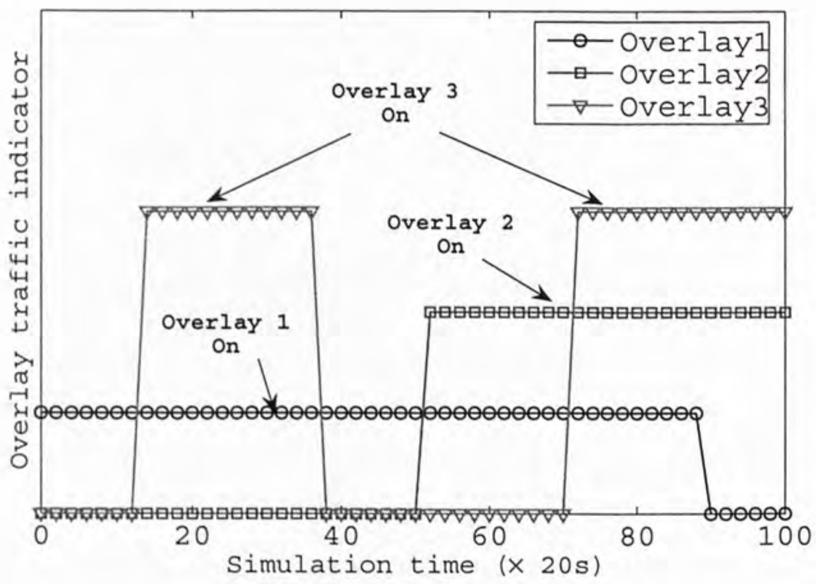


Figure 6.5: Traffic dynamics (each unit in the x-axis is 20 sec)

launches its traffic transmission from $N0$ to $N8$ and this transmission terminates at $t = 1,800$. At $t = 1,000$, overlay 2 launches its traffic from node $N1$ to $N10$ and this transmission continues until the end of the simulation. The traffic of overlay 3 begins at $t = 250$ with source node $N2$ and destination node $N7$. This transmission terminates at $t = 750$ but restarts at $t = 1,400$ and continues till the end of the simulation.

Figure 6.6 indicates the instantaneous throughputs of links $N7-N8$, $N7-N10$ and $N8-N10$, which are some of the common links shared by the overlays. We observe that the throughputs of all three links increase at around $t = 250$ because of the arrival of new traffic from overlay 3 and decrease accordingly at its departure at $t = 750$. Similarly, one can observe the corresponding increase and decrease of link throughput as the traffic dynamics occur along the time line.

Figure 6.7 shows how each overlay's routing decision adapts to the network dynamics. We observe that at time intervals when traffic is stationary, the rate

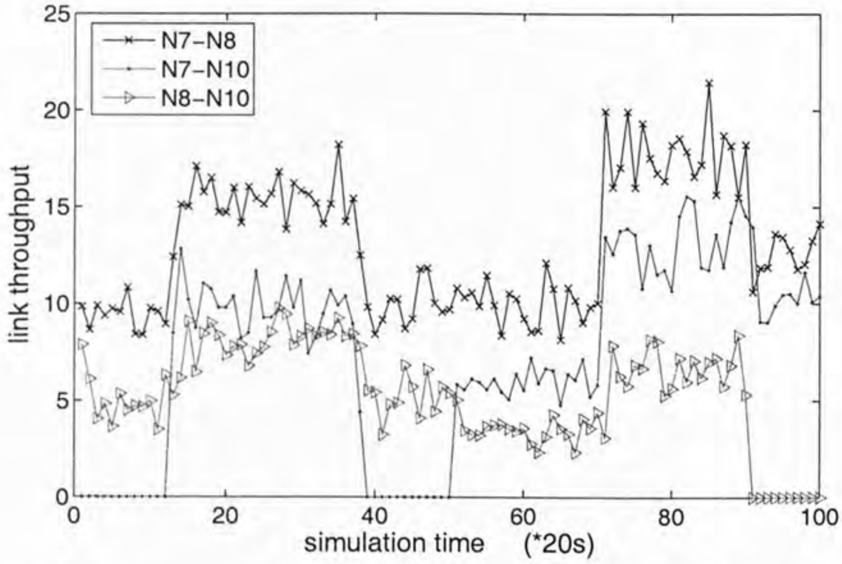


Figure 6.6: Instantaneous link throughput v.s. simulation time (each unit in the x-axis is 20 sec)

of traffic assigned to different paths for an overlay remain constant and slowly converge to the equilibrium. When traffic enters/departs, the routing decision of overlays adapts quickly and reach a new equilibrium point. This confirms our theoretical claim of the convergence to the Nash equilibrium. One also notes that for the packet based simulation, unlike the fluid-based simulation, there seem to be some fluctuations at the equilibrium point. This is due to variation in packet sizes and burstness in packet arrival times in the simulation. We found that if we use a CBR as the traffic source, the results are very similar to that of fluid-based simulations.

Figure 6.8 compares the performances achieved by different overlays as a function of time when doing the simulation. In the left graph, we plot the average delay ratio of overlay 3 to that of overlay 2. Note that the average delay in overlay 3 is always larger than overlay 2. However, overlay 2 temporarily achieves a better performance due to the departure of traffic in overlay 1. At this point, we find that both overlay 2 and 3 achieve *lower* average delays than what they experienced in the subsequent equilibrium, which again verifies

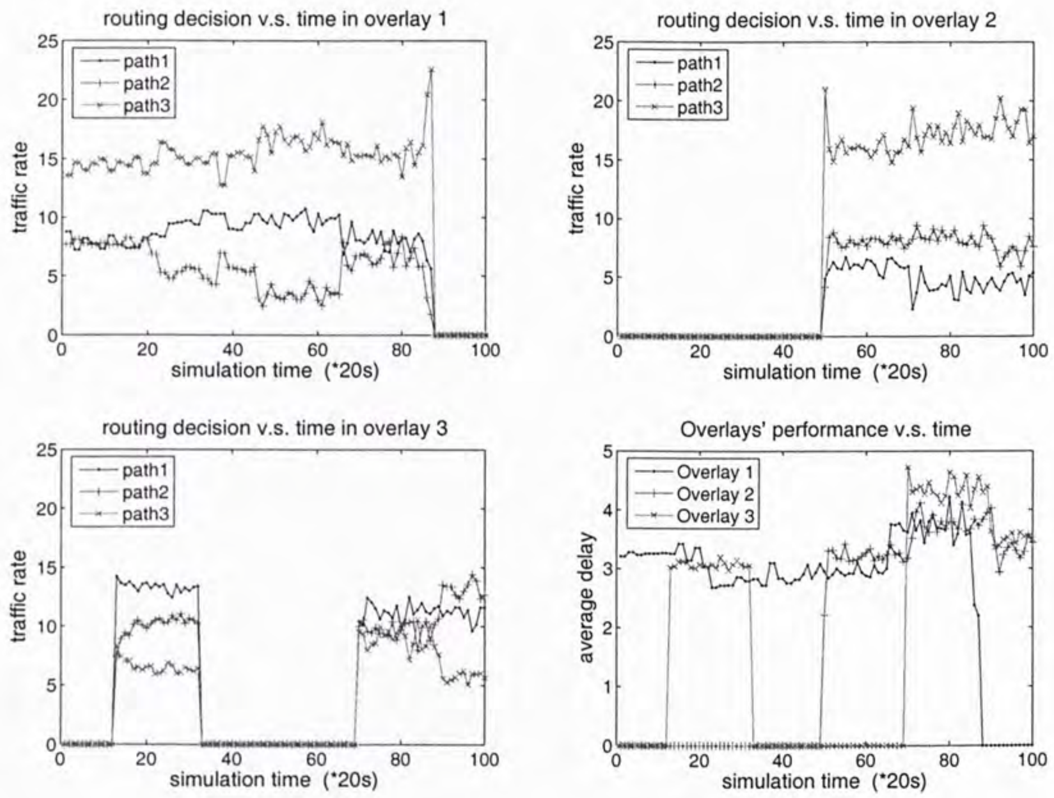


Figure 6.7: Routing decisions of overlays and performance comparison (each unit in the x-axis is 20 sec)

that the Nash equilibrium is not Pareto optimal. In the right most graph, we plot the ratio of the average delay ratio of overlay 2 to that of overlay 1. Again, one observes that the variation in the performance ratio due to the rejoin of traffic in overlay 3. In this case, it is not fair for overlay 2 at the equilibrium to always bear a larger delay than that of overlay 1. This confirms that the fairness paradox exists in a realistic network setting. In summary, this shows the unfairness of resource allocation of the common resource (i.e., shared links).

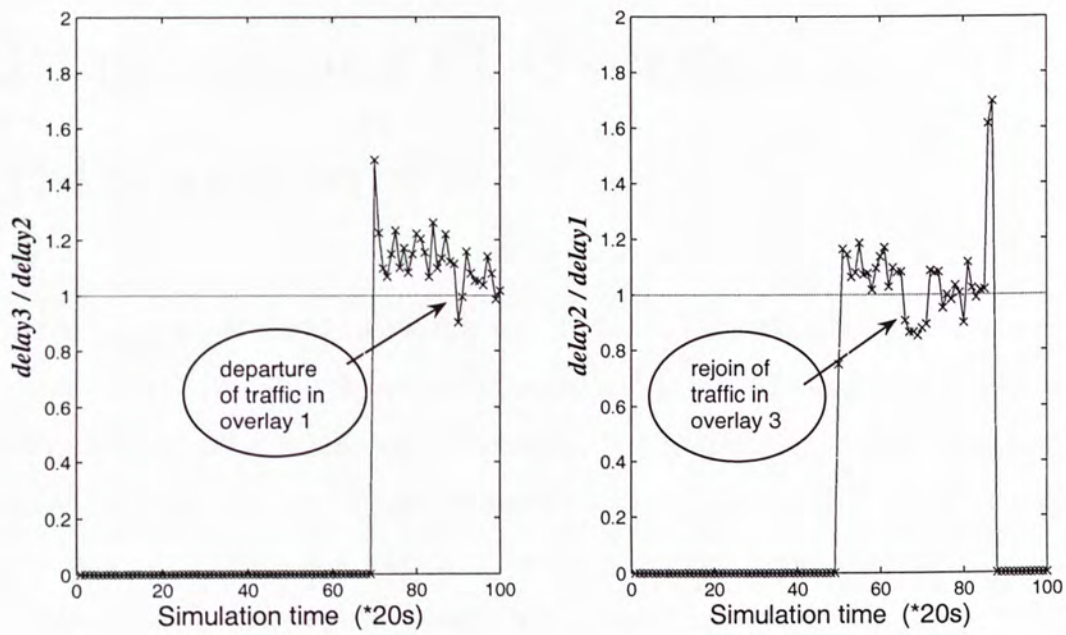


Figure 6.8: Performance comparison v.s. network dynamics (each unit in the x-axis is 20 sec)

Chapter 7

Understanding Various Issues & Implications of Overlay Interaction

In this section, we discuss some intrinsic problems that arise due to interactions between overlays. These include sub-optimality in performance and certain fairness anomalies in resource allocation. It is important to point out that these problems are not unique to overlay optimal routing policy, but rather, common to all forms of application layer routing that have interaction among overlays. Since many overlays are now appearing in the Internet, unregulated application routing may degrade the performance of all users. Worse yet, because overlays may not realize the existence of other overlays, these problems will persist due to the convergence to the equilibrium point.

Let us use an example to illustrate these issues. A physical network consisting of six nodes is depicted in Figure 7.1. There are two overlays in the network: overlay 1 consists of nodes A, C, E while overlay 2 consists of nodes B, D, F. For overlay 1, all logical links map to the corresponding physical links except for the logical link between node C and node E, which corresponds to the physical links $C - D - E$. For overlay 2, all logical links map to the corresponding physical links except for the logical link between node B and node

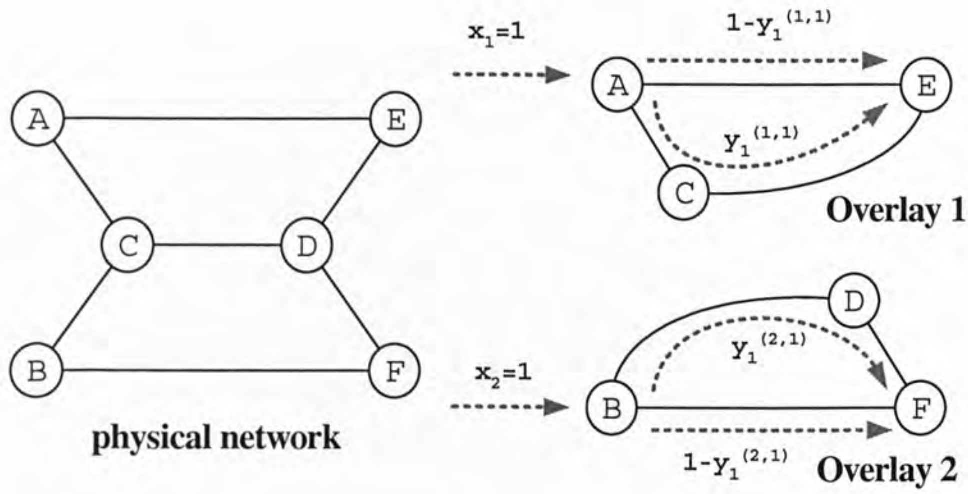


Figure 7.1: A simple network with two overlays to illustrate potential problems

D, which corresponds to the physical links $B - C - D$. Thus, physical link $C - D$ is the common link *shared* by these two overlays.

7.1 Sub-optimality of Nash Equilibrium

Assume that both overlays have one source-sink pair and one unit of traffic demand: $x_1 = x_2 = 1.0$. We define the following delay functions for various physical links in the physical network: $d_{A,E}(y) = a + y$; $d_{C,D}(y) = by^{\alpha}$; $d_{B,F}(y) = c+y$, while other links have zero delay. Here, y represents the aggregate traffic traversing a link, and a, b, c, α are some non-negative parameters of the delay functions.

Let us consider the routing decisions of these two overlays. Overlay 1 routes $y_1^{(1,1)}$ units of traffic through logical path A-C-E and $(1-y_1^{(1,1)})$ (because $x_1 = 1$) units of traffic via logical path A-E. On the other hand, overlay 2 routes $y_1^{(2,1)}$ units of traffic through logical path B-D-F and $(1 - y_1^{(2,1)})$ units of traffic via B-F. Similarly, the KKT condition for overlay 1 is

$$a + 2\left[1 - y_1^{(1,1)}\right] = b\left[y_1^{(1,1)} + y_1^{(2,1)}\right]^{\alpha} + y_1^{(1,1)} \cdot b\alpha\left[y_1^{(1,1)} + y_1^{(2,1)}\right]^{\alpha-1} \tag{7.1}$$

while the KKT condition for overlay 2 is

$$c + 2\left[1 - y_1^{(2,1)}\right] = b\left[y_1^{(1,1)} + y_1^{(2,1)}\right]^{\alpha} + y_1^{(2,1)} \cdot b\alpha\left[y_1^{(1,1)} + y_1^{(2,1)}\right]^{\alpha-1}. \tag{7.2}$$

where $y_1^{(1,1)}, y_1^{(2,1)} \in [0,1]$.

One can easily show that in the overlay optimal routing game described above, the Nash equilibrium point is *not* Pareto optimal. A Pareto optimal point is defined as a strategy profile for all overlays such that no overlay can use another routing strategy that can decrease its own weighted average delay *without* increasing other overlays' weighted average delay. Although all overlays perform an individual optimization at every routing update and the system will finally reach a Nash equilibrium point, the equilibrium may be inefficient since there exists another routing strategy at which *all* overlays can achieve a better performance than at the Nash equilibrium.

To show the sub-optimality of the Nash equilibrium in the optimal routing game for the network depicted in Figure 7.1, we consider the KKT conditions specified by Equations (7.1) and (7.2). Assume we have the following parameters for the delay functions: $\alpha = 1, a = 1, b = 1$ and $c = 2.5$, one can simply verify that the Nash equilibrium in this example is $\{y_1^{(1,1)} = 0.5, y_1^{(2,1)} = 1\}$, that is, overlay 1 uses both paths while overlay 2 uses a single path, which consists of the shared link. The weighted average delays for both overlay 1 and 2 are 1.5. However, if we consider another routing strategy profile of $\{y_1^{(1,1)} = 0.4, y_1^{(2,1)} = 0.9\}$, one can find that the weighted average delays for overlay 1 and 2 are 1.48 and 1.43 respectively, which are *lower* than the delay achieved at the Nash equilibrium.

7.2 Slow convergence to Nash equilibrium

The second problem in this competitive routing game regards the rate of convergence to the Nash equilibrium. Although we show in the previous section the existence of a Nash equilibrium under general network settings, this does not ensure how fast the equilibrium point can be reached. To show how the convergence rate is affected by the capacity of shared links, we use the network in Figure 7.1 to illustrate the relationship. Here, assume that the delay function of each link is represented by an M/M/1 model. Both overlays have a traffic demand of $x_1 = x_2 = 5$ Mbps, and the capacities of various links are: $C_{AC} = C_{EC} = C_{BD} = C_{FD}$=10 Mbps, $C_{AB} = 6$ Mbps and $C_{EF} = 5$ Mbps. Now we vary the capacity of link CD, which is the only shared link in this example, from 8 to 6 Mbps. From Figure 7.2, one can observe that when the capacity is 8 Mbps, two overlays quickly reach an equilibrium. As we reduce the capacity of link CD, it takes longer to reach the equilibrium. In the right sub-figure of Figure 7.2, one can observe that the settling time is much longer, *i.e.*, traffic flipping back and forth between different routes and takes longer time to stabilize. In fact, when the traffic through some path decreases, there is a corresponding increase in another path, which results in an oscillation-like scenario. One can interpret that when the aggregate traffic rate is approaching the link's capacity, overlays will experience a very large delay and thus results in a highly varied traffic allocation. Indeed, this is one main cause of the slow convergence rate.

7.3 Fairness Paradox

Another more severe problem is the notion of fairness in resource allocation. We use the same network in Figure 7.1 to illustrate the problem. Note that these two overlays are symmetric, each having two paths: a shared path and a private path. Although overlay 2 is "*worse off*" by having a private path (link $B - F$) with higher delays than that of overlay 1's private path (link $A - E$), as in the previous example, it is able to achieve the same average delay as overlay 1 in the Nash equilibrium. This is because overlay 2 is able

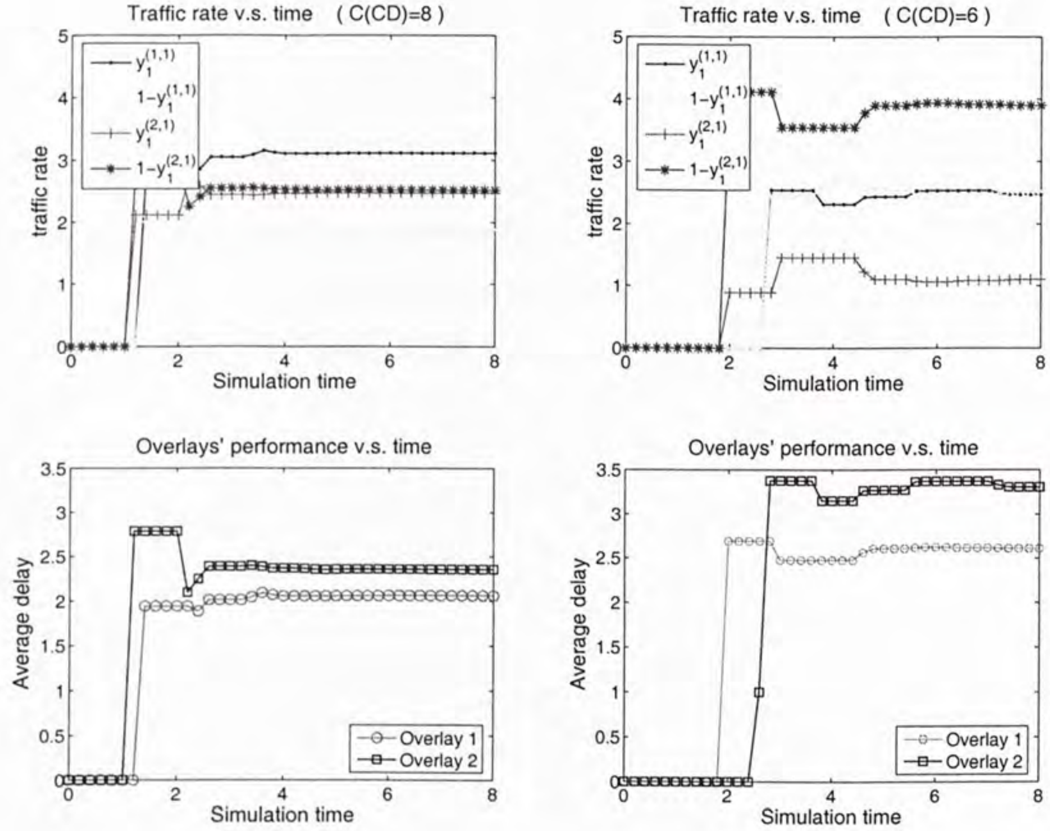


Figure 7.2: Convergence of traffic rate and delay as we vary the capacity C_{CD}

to fully take advantage of the lower delay of the shared path, whereas it only makes sense for overlay 1 to send part of its traffic over the shared link due to its superior private path. In fact, one can find delay functions such that the situation is *arbitrarily worse*.

To illustrate, note that the delay function for the shared link $C-D$ is $d_{C,D}(y) = by^{\alpha}$. One can ask for what values of a and c, which are the parameters of the delay functions for the private link of overlay 1 and 2, do the Nash equilibrium solution remain at $\{y_1^{(1,1)} = 0.5, y_1^{(2,1)} = 1\}$? Such values for a and c exist, in particular, when $b = 1$ and $a < c$, we have:

$$a = \left(\frac{3}{2}\right)^{\alpha} + \frac{\alpha}{2}\left(\frac{3}{2}\right)^{\alpha-1} - 1 \quad ; \quad c = \left(\frac{3}{2}\right)^{\alpha} + \alpha\left(\frac{3}{2}\right)^{\alpha-1}.$$

In Section 7.1, we showed when $a < c$ and $\alpha = 1$, $\text{delay}_1 = \text{delay}_2$. When $\alpha > 1$:

$$\text{delay}_1 = \left(\frac{3}{2}\right)^{\alpha} + \frac{\alpha}{4}\left(\frac{3}{2}\right)^{\alpha-1} - \frac{1}{4}; \qquad \text{delay}_2 = \left(\frac{3}{2}\right)^{\alpha},$$

and observe now that delay_1 becomes *greater* than delay_2. This implies that overlay 2 is able to achieve *better* performance despite starting with a *worse* private link in an otherwise symmetric situation with overlay 1. Furthermore, as we increase α, this unfairness can even be *unbounded*, that is:

$$\left.\frac{\text{delay}_1}{\text{delay}_2}\right|_{\alpha\to\infty} = \infty$$

and this is depicted in Figure 7.3(a).

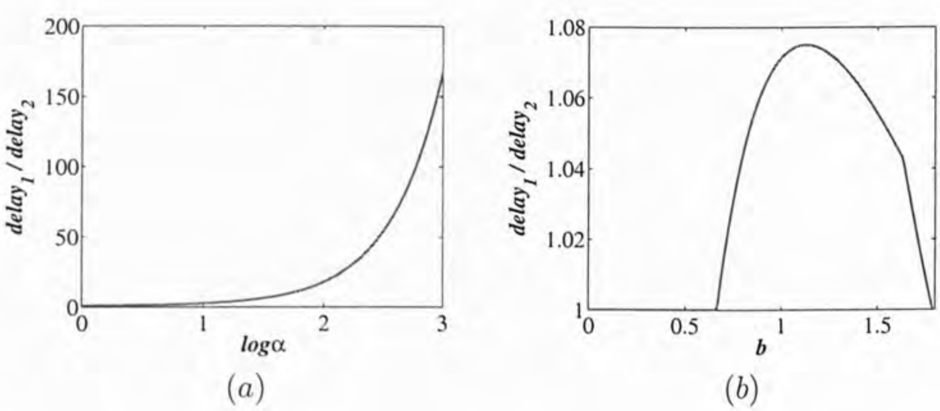


Figure 7.3: $delay_1/delay_2$ ratio v.s. (a) $\log(\alpha)$: unfairness becomes unbounded; (b) parameter b with $a = 2, c = 4$ and $\alpha = 1$: bounded unfairness.

This type of anomaly also exists for other sets of parameters, for example, when $a = 2, c = 4, \alpha = 1$ and we vary the values of b, one can observe that overlay 1 will have a worse performance compared to overlay 2 (though the unfairness is bounded in this case). In summary, we illustrate that there exist delay functions for the links such that although overlay 1 seemingly has better paths than overlay 2, it is destined to lose the routing game to overlay 2 by an arbitrary margin: a rather "*paradoxical*" situation.

Chapter 8

Overlay Pricing

In Chapter 7, we observed some undesirable properties achieved at the equilibrium due to unregulated competition between overlays for the common resources. In this section, we explore some pricing schemes to address these potential problems, namely (a) sub-optimality at the Nash equilibrium and (b) fairness anomaly issues. We show the efficiencies of our pricing schemes and give their economic interpretations.

8.1 Pricing mechanism to improve end-to-end delay

In Chapters 7 and 5, we showed that the Nash equilibrium point of the overlay optimal routing game is not Pareto optimal, thus bearing a performance gap to the global optimum. In this chapter, we show that, by incorporating proper pricing schemes, an ISP can bring the equilibrium of the overlay routing game to the global optimum. The implication of this result is that not only the performance gap can be bridged, but at the same time, this provides an extra revenues and incentive for the ISP to implement the pricing strategy.

The basic idea of the pricing scheme is to introduce a price (cost per unit traffic) for every physical link in the network. When overlays make the routing decision, they need to consider the average delay as well as the total price they pay to the network operators for consuming the link bandwidth. Formally, there is a load function $U_s(y^{(s)})$ associated with each overlay, which the overlay aims to minimize. With the prices set by the links, each overlay, say s, needs to determine its routes by solving the following optimization problem:

$$\begin{aligned} \text{Minimize} \quad & U_s(y^{(s)}) = \alpha_s \cdot delay^{(s)} + payment^{(s)} \\ \text{where} \quad & payment^{(s)} = \textstyle\sum_{j \in \mathcal{J}} l_j^{(s)} \cdot p_j^{(s)}. \end{aligned} \tag{8.1}$$

Here $delay^{(s)}$ is the average delay in overlay s defined in Equation (3.4) and $payment^{(s)}$ accounts for the total payment of overlay s for consuming the link bandwidth for all traversed links. Note that here α_s can be interpreted as the *sensitivity factor*, which is determined by the overlays themselves to reflect the overlay's own preference between performance and the cost. Therefore, for those overlays who desire a better performance, they will set α_s to a larger value. With proper translation, the total payment can be summed over all traversed links, wherein each link has a charge of $l_j^{(s)} \cdot p_j^{(s)}$.

Note that here we adopt a price-discriminating strategy: ISPs will charge different prices for different overlays. Therefore, the unit price of each link is different for different overlays. Formally, for each link j, we set a price $p_j^{(s)}$ for overlay s. With the previous definition for the average delay, the routing optimization problem (8.1) for overlay s can be rewritten as:

$$\text{Minimize} \sum_{j \in \mathcal{J}} \alpha_s \cdot l_j^{(s)} \cdot d_j(l_j^{(s)} + l_j^{(-s)}) + l_j^{(s)} \cdot p_j^{(s)} \tag{8.2}$$

wherein $l_j^{(s)}$ denotes the traffic of overlay s on link j, and $l_j^{(-s)}$ denotes other traffic (from the underlay and other overlays) on link j. The key idea of the pricing scheme is that by incorporating the pricing mechanism, we introduce a penalty for over-utilizing some links. If the price $p_j^{(s)}$ is set properly to reflect the economic relationship with the delay metric, one can gain a balance between overlays' performance and the payment. The following theorem shows how the price can be set so that the equilibrium point equals the global optimum.

Theorem 8.1 *In the overlay optimal routing game* G_{overlay} *if (a) the objective function of each overlay is set as in Eq. (8.2) and (b)* $p_j^{(s)} = \alpha_s \cdot l_j^{(-s)} \cdot d_j'(l_j)$*, where* $l_j = l_j^{(s)} + l_j^{(-s)}$*, the Nash equilibrium achieves the same performance as the global optimal routing.*

Proof: The objective of the global optimal routing is to minimize the weighted average delay for the overall traffic in the network, which is $\sum_{j \in \mathcal{J}} l_j \cdot d_j(l_j)$, where l_j is the total traffic on link j. Thus the KKT condition for the optimal point achieved by the global optimal routing can be written as follows: for every route r with non-zero traffic, and another route r' serving the same source-sink pair (or *flow*) f, we have:

$$\sum_{j \in r} [l_j \cdot d_j(l_j)]' = u_f \leq \sum_{j \in r'} [l_j \cdot d_j(l_j)]'.$$

On the other hand, the KKT condition for the optimization problem defined in Eq. (8.2) requires that for every route r with non-zero traffic, and another route r' serving the same flow f in overlay s, must satisfy:

$$\begin{aligned} & \textstyle\sum_{j \in r} \left[\alpha_s \cdot l_j^{(s)} \cdot d_j(l_j^{(s)} + l_j^{(-s)}) + l_j^{(s)} \cdot p_j^{(s)}\right]' = u_f \\ \leq \; & \textstyle\sum_{j \in r'} \left[\alpha_s \cdot l_j^{(s)} \cdot d_j(l_j^{(s)} + l_j^{(-s)}) + l_j^{(s)} \cdot p_j^{(s)}\right]' \end{aligned}$$

If the price is set $p_j^{(s)} = \alpha_s \cdot l_j^{(-s)} \cdot d_j'(l_j)$, we have

$$\begin{aligned} & \textstyle\sum_{j \in r} \left[\alpha_s\Big(d_j(l_j) + l_j^{(s)} \cdot d_j'(l_j)\Big) + p_j^{(s)}\right] &= u_f \\ \Leftrightarrow \; & \textstyle\sum_{j \in r} \left[\alpha_s\Big(d_j(l_j) + l_j^{(s)} \cdot d_j'(l_j)\Big) + \alpha_s \cdot l_j^{(-s)} \cdot d_j'(l_j)\right] &= u_f \\ \Leftrightarrow \; & \textstyle\sum_{j \in r} \alpha_s \big[d_j(l_j) + l_j \cdot d_j'(l_j)\big] &= u_f \\ \Leftrightarrow \; & \textstyle\sum_{j \in r} \alpha_s \big[l_j \cdot d_j(l_j)\big]' &= u_f \end{aligned}$$

which proves that under this pricing scheme the Nash equilibrium point for the overlay routing game has the "*same*" KKT condition as the global optimal routing, therefore achieving an globally optimal performance. ■

Table 8.1 illustrates the distributive algorithm for implementing the overlay optimal routing policy incorporating the discriminate pricing scheme. Here we assume overlays are able to obtain the necessary routing information from a common routing underlay [12] and each link (router) is able to differentiate the traffic source so that they can set a discriminate price for different overlays. Doing each routing update, the traffic sources in an overlay network send probes to all available logical paths. Every physical link along a logical path processes the routing probe, and puts the information tuple $\langle j, l_j, p_j^{(s)}, C_j \rangle$ into the probe packet. Collecting the information carried by the acks of probe packet, overlays can determine their optimal routing strategy by solving the optimization problem in Eq.(3.5).

8.1.1 Fluid-level Simulation

Here, we present the simulation results on both fluid basis and packet basis, and to show the effectiveness of our pricing scheme.

For the fluid-based simulation, we use the same network topology depicted in Figure 6.1. The same traffic demand for each overlay and the link capacities are also adopted. Figure 8.1 depicts the routing decisions under the pricing scheme. Note that after incorporating the pricing scheme, the flow allocation among different paths has higher variation, compared with the situation without pricing. This is due to the more intensive interactions between overlays, since the price for an overlay depends heavily on the routing decisions of other overlays. Nevertheless, convergence can still be achieved.

Table 8.2 gives the performance comparison (average weighted delay) under different routing policies, namely, the overlay optimal routing without/with pricing and the centralized global optimal routing. Some observations are

---

▷ For each physical link j in the underlay:

Repeat Determine that the current routing probe is from overlay s and its sensitivity factor α_s.
Record the instantaneous link throughput l_j.
Record the instantaneous link throughput $l_j^{(s)}$ whose traffic originates from overlay s.
Calculate $l_j^{(-s)} = l_j - l_j^{(s)}$ and the price for overlay s equals $p_j^{(s)} = \frac{1}{\alpha_s} \cdot l_j^{(-s)} \cdot d_j'(l_j)$.
Tab the tuple $\langle j, l_j, p_j^{(s)}, C_j \rangle$ to the probe packet.

▷ For each overlay s:

Repeat Assume each overlay is able to obtain the routing information , e.g., topology about the underlying network, from a common routing underlay.
Send the routing probe to each logical overlay path.
Collecting information $\langle j, l_j, l_j^{(-s)}, C_j \rangle$ on each link j on the overlay path.
Determine routing by solving the optimization problem in Eq.(3.5).

Table 8.1: Distributive algorithm for overlay optimal routing with discriminate pricing mechanism

Overlay	1	2	3	4	5	6	Overall
Overlay optimal routing	9.53	4.58	10.57	8.37	10.77	0.33	44.15
Overlay routing (pricing)	9.93	4.40	8.67	8.40	9.73	0.32	41.45
Global optimal routing	9.92	4.40	8.66	8.41	9.72	0.33	41.44

Table 8.2: Comparison of weighted average delays under different routing strategies

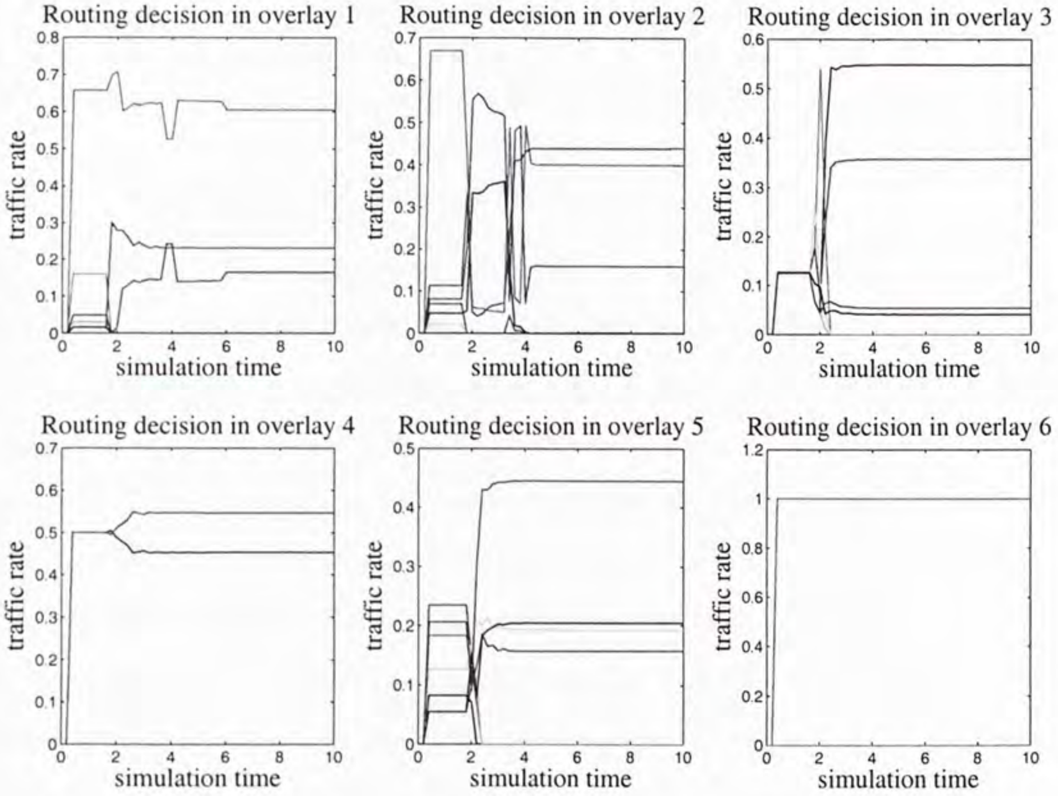


Figure 8.1: Routing decisions for six overlays with pricing scheme

made here. The pricing mechanism achieves close-to-optimum performance, and reduces the overall delay as compared to the scheme without pricing. Note that after incorporating the pricing scheme, though the average delay in some overlay is increased, the delays in some other overlays decrease at a higher rate, which leads to the global optimum. This can be interpreted that the marginal cost pricing brings in extra penalty for the increased delay burdened on other overlays, caused by the overlay's own traffic. Table 8.3 tabulates the total payment for different overlays when their routing behaviors reach an equilibrium. Note that when an overlay shares more congested physical links with other overlays, the more payment it has to bear. In our simulation, overlay 6 has only one logical path which shares some common links with other overlays, thus bearing the lowest payment.

Overlay	1	2	3	4	5	6
Payment	26.5	8.6	15.8	31.5	37.9	0.01

Table 8.3: Overlays' payments under the discriminate pricing scheme.

8.1.2 Packet-level Simulation

For the packet-based simulation, we repeat the same simulation scenario as presented in Chapter 6, except that each overlay receives an additional price as we have suggested. In Figure 8.2, we plot the routing decisions of three overlays under the pricing scheme, as well as their performance comparison. Note that similarly, every flow allocation dynamic corresponds to the entering/leaving of the overlay traffic. Also note that the traffic variation is comparably larger than the situation without pricing, which verifies the explanation presented in the fluid-based simulation. To compare the performance gap reduced by the pricing mechanism, we plot the *instantaneous* percentage of the improved performance (reduced delay) in Figure 8.3. In here, the reduced delay refers to the difference of the overall delay in three overlays with and without pricing, and we compare it to the performance with pricing as the improved percentage. We numerically show that the average performance improved is around 5% in this example. Though the performance gap is not distinct here, we argue that it does not weaken the strength of the pricing mechanism. The performance gap can be arbitrarily large for arbitrary link delay functions and network topologies.

8.2 Pricing mechanism to improve fairness

One important incentive for the network operators (ISPs) to perform network upgrade is to increase profit. Furthermore, it is desirable if the pricing and revenue sharing schemes correctly represent the contribution of different ISPs, while providing fair resource allocation to the users in the network. In the

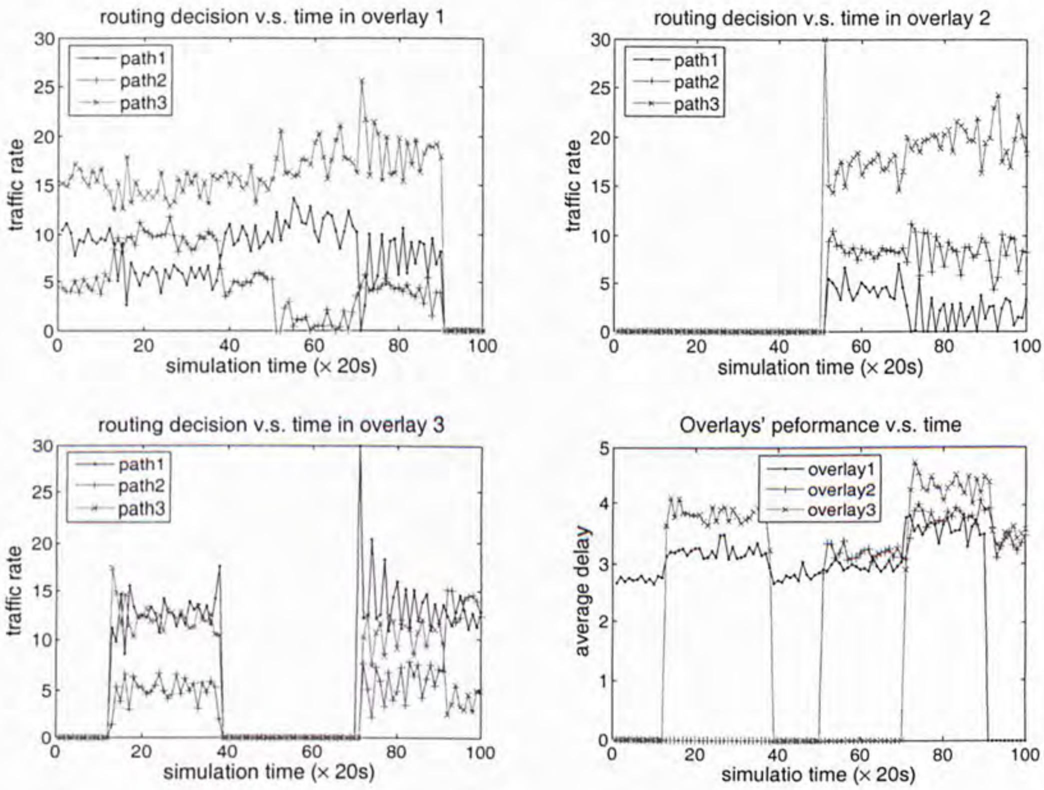


Figure 8.2: Routing decisions of overlays and performance comparison under discriminate pricing

previous section, we have observed some fairness anomalies due to unregulated competition between overlays. Worse yet, such unfairness can even be unbounded for general situations. Here, we explore another pricing scheme that can achieve a spectrum of fairness balanced between the performance and the cost, and reflects the economic benefit for the network operators.

Our pricing scheme is based on the following two natural arguments. Firstly, pricing should reflect users' willingness to pay, meaning that those users who pay more have greater opportunity to receive better performance. Secondly, the network operators set prices to maximize their own economic interests. Before proceeding to the formal pricing model, we make some additional assumptions, for both simplicity and ease of presentation. We assume that those

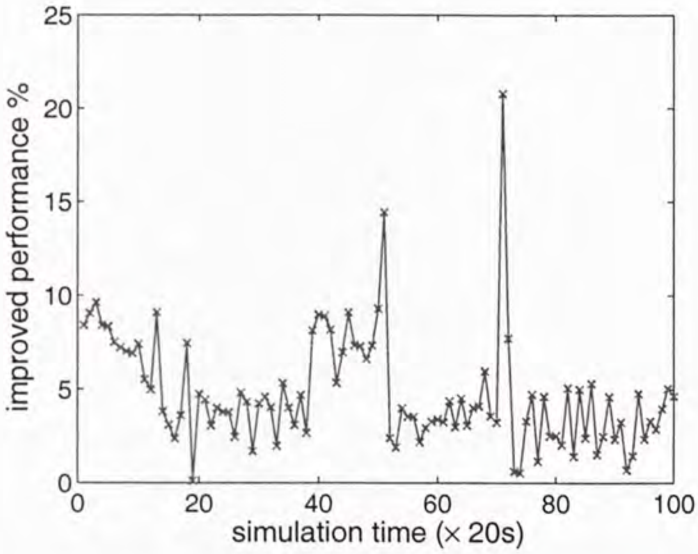


Figure 8.3: Instantaneous percentage of improved performance by incorporating discriminate pricing mechanism

"common" physical links in the network represent network operators that have their own economic interests. The argument is that an overlay network may reside over many ISPs and these physical links constitute the bottleneck links within one ISP. This is because one provider's egress traffic is often charged by its downstream providers and these bottleneck links often get saturated before other links do.

Under this assumption, the pricing scheme can be decomposed into two components, one for the overlay users and the other for the network operators. For each physical link, it is interpreted as a logical ISP which tries to maximize its own economic benefit. Each link j sets a price p_j, which is the charge for per unit traffic traversing this link, to maximize its own profit. The profit of a link j is defined as $P_j(l_j) = p_j \cdot l_j - c_j(l_j)$, where the first part accounts for the revenue received by the link, while the latter part $c_j(l_j)$ is interpreted as the operating cost at current congestion level. Note that a link sets the same price for *all* overlays and there is no price discriminate. Formally, each link j

solves the following optimization problem:

$$\text{maximize} \quad P_j(l_j) = l_j \cdot p_j - c_j(l_j) \tag{8.3}$$

To maximize the profit, link j would not set the price arbitrarily large, otherwise no overlay would choose to traverse this link. Given a fixed price p_j, the optimal aggregate traffic on a link j to maximize the profit can be obtained by solving $dP_j/dl_j = p_j - c_j'(l_j) = 0$, thus having $p_j = c_j'(l_j)$. To match the actual bandwidth consumption demand, link j will set its price by $p_j = c_j'(l_j)$. Note that the operating cost $c_j(l_j)$ is a non-decreasing function of the aggregate traffic on the link. We argue that this is an approach for links to estimate the traffic demand on them. This equivalently determines the supply curve for the link resources.

Associated with each overlay is a load function $U_s(y^{(s)}; y^{(-s)})$ that it wants to minimize. With the prices set by the links, one overlay s determines its routing by solving the following optimization problem:

$$\text{minimize} \quad U_s(p^{(s)}; p^{(-s)}) = \alpha_s \cdot delay^{(s)} + \sum_{j \in \mathcal{J}} l_j^{(s)} \cdot p_j \tag{8.4}$$

wherein the definition is similar to Eq.8.2. Also note that the constraints to this optimization problem is the same as the overlay optimization given in Eq.(3.5). By writing down the KKT conditions to this optimization problem, it is easy to show that the overlay routing game under objective function in Eq. (8.4) has a Nash equilibrium when p_j is considered as a fixed parameter by overlays. The routing decisions made by the overlays actually determine the demand curve of the link resources (bandwidth).

To see how this pricing scheme works, the cost function of link j can be set in the following way: $c_j(l_j) = \log[l_j \cdot d_j(l_j)]$, such that the cost of operating a link increase logarithmically with the weighted delay on that link. In other words, $l_j \cdot d_j(l_j)$ can be interpreted as the total "*bits*" carried by this link.

Therefore, the price of a link is determined:

$$p_j = c_j'(l_j) = 1/l_j + d_j'(l_j)/d_j(l_j) \tag{8.5}$$

wherein the second part can be interpreted as the *elasticity* of the delay function of link j. In Figure 8.4, we plot the price-traffic curve for different delay functions. In here some observation is made. Firstly, the price function is a decreasing function of the traffic traversing it. The justification is that when *less* traffic traverses a link, the link bears a *lower* delay, which implies a *higher* price. Therefore for overlays that choose links with lower delay to ensure a better performance, they have to pay more to buy the service. Note that in Figure 8.4(c), though the price tends to be an increasing function when the traffic rate exceeds half of the capacity, the high delay when the rate approaches the capacity prevents overlays from using this link.

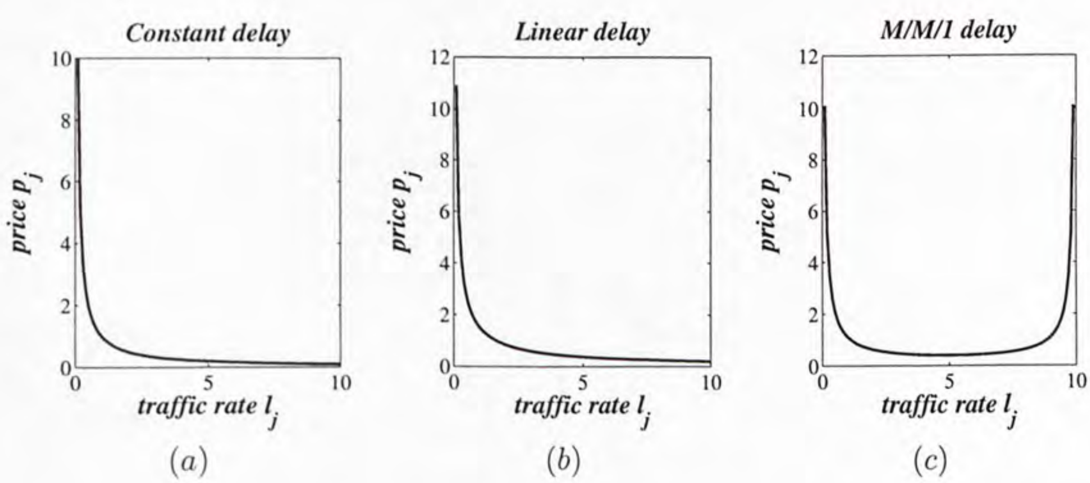
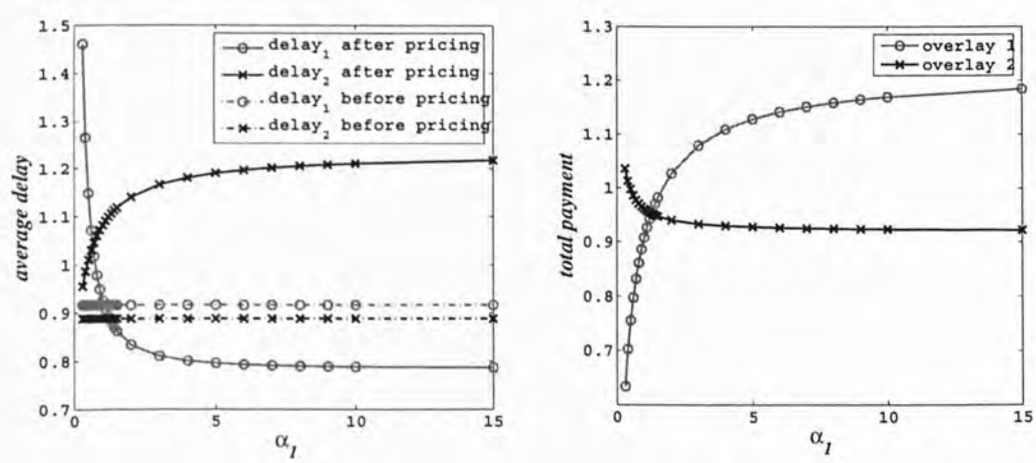


Figure 8.4: Price-traffic curves for links with different delay functions, with link capacity $C_j = 10$ units.

We use the same network topology illustrated in Figure 7.1 to compare the performance before and after incorporating our pricing scheme. In here, we assume that links $A-E$ and $B-F$ (with capacity $C_{AE} = 2$, $C_{BF} = 1.5$) are the "*private*" links of overlay 1 and overlay 2 respectively, and they do not have to pay for them. The only common physical link is $C-D$ with capacity $C_{CD}=6$ and overlays have to pay for consuming the bandwidth on this link. For the ease of presentation and analysis, we assume other links have sufficiently large capacities and do not charge on the two overlays. In Figure 8.5, we show the performance comparison before and after taking the pricing scheme. Overlay 2's sensitivity factor α_2 is kept fixed, while α_1 is changed to see the different equilibrium points under this pricing. Some observation is made. Before taking the pricing scheme, overlay 1 achieves a higher delay than overlay 2, despite owning a "*better*" private link in this otherwise symmetric environment. After bringing in the pricing scheme, overlay 1 achieves better delay as α_1 increases, at the cost of more payment to link $C-D$. However, one can observe that there still exists intervals of α_1 such that overlay 1 achieves better delay than overlay 2, yet still bearing a less payment. This is not surprising since overlay 1 owns a better private link.



(a) (b)

Figure 8.5: Comparison of equilibrium point before and after taking pricing scheme ($\alpha_2 = 0.5$).

Chapter 9

Related Work

In here we provide an overview of the related work that motivates our work or parallels our study on similar topics. Though the study on the interaction of co-existing overlay networks is receiving some attention, it has not been well addressed up to date.

Routing scheme which achieves global optimality was well studied in [65]. Such a "global optimal routing" achieves a minimum average latency for *all* the traffic in the network by solving a centralized optimization problem. Since traditional Internet routing is not load dependent, but rather, policy-based across multiple ISPs, therefore, Savage *et. al.* [10] have observed an inherent inefficiency of network-level routing from the users' perspective and showed that the default Internet path is often suboptimal in terms of delay, loss rate and bandwidth. Such a sub-optimality is inevitable due to the different objectives of network operators and end-to-end users, as well as different network topologies and routing policies. However, with the increasing popularity of overlay networks (i.e., RON [2]), and the possibility of source routing like Detour [9], there exists an opportunity to allow end users to choose application-specific routes, i.e., routes with the shortest delay (or smallest loss rate, maximal available bandwidth). This is known as the *selfish routing*.

Roughgarden [11] pointed out that the network will experience a performance degradation if each end user performs selfish routing and also gave the theoretical upper bound for this efficiency loss. However, recent experiments in real network environments [62] showed that selfish routing achieves a close-to-optimal performance in terms of average delay, but at the cost of overloading some links, wherein there is a contradiction to the theoretical results. Xie *et. al.* [63] proposed a probabilistic routing scheme to implement the selfish routing in dynamic environments, ensuring that the selfish behaviors of end users will converge to a routing *equilibrium* that resembles what is studied in the theoretical work [11]. In addition, the proposed routing scheme shows a promise for an overlay or an end-user to implement the optimal routing, which selfishly maximizes the entity's own utility but exposes the possibility of degraded performance experienced by other users or overlays.

The seminal work by [48] presented the first attempt of a game-theoretic analysis of interaction between one overlay network and the underlay (physical) network. The overlay performs application layer routing and tries to minimize its average delay. At the same while the ISP performs traffic engineering so as to minimize the average delay for all traffic within the physical network. They showed the equilibrium point for a simple example and explored some interesting implications in the example network. Unlike their work, we consider the interaction of multiple overlays. Not only do we show the existence of an equilibrium in generic networks, but we also discover some potential anomalies and study how the performance gap can be bridged by a distributed pricing scheme.

The interaction between co-existing overlay networks is also studied in [45–47], which differed from our work in another perspective of focus. One common characteristic of these work is that they investigated the simultaneous responses of multiple overlays to a routing incidence, for instance, the congestion or failure of a highly loaded link. Authors showed that for some particular network topologies and traffic patterns, the load-sensitive routing (i.e., always selecting a path with the shortest delay) can cause persistent routing oscillation. Though the possibility of oscillation exists, it can be simply eliminated by taking the advantage of a probabilistic routing scheme [63].

Last but not the least, the ability of an overlay network to compensate the carelessness of the underlay routing scheme was studied in [67]. Other documents that explored the undesirable effects due to the interaction between co-existing overlay networks were reported in [68, 69].

Chapter 10

Conclusion

The central motif of this thesis is the investigation of the interaction between multiple co-existing overlay networks, and the inherent implications reflected by the interaction. With the discovery of some anomalies that may exist in a routing equilibrium, we analyze how this lack of coordination affects the system stability, performance as well as fairness in resource allocation. This thesis only represents one corner of a problem which requires a multifaceted approach. In this chapter, we are going to summarize our contributions and briefly discuss some key open issues on which further study is necessary.

10.1 Summary of the Contribution

Throughout this thesis, we consider the co-existence and interaction of multiple overlays on top of a physical network. We consider the situation wherein overlays can determine their routing so as to optimize their individual performance measures. A constrained convex optimization framework is utilized to formulate the selfish (user-optimal) routing and overlay optimal routing policy. In particular, we present a non-cooperative game framework to characterize overlays' behaviors. We show the existence of a Nash equilibrium in a generic network setting, which implies that there will be no routing instability.

However, through some simple illustrations, we demonstrate some inherent anomalies due to these interactions, namely, degraded performance measure and some fairness paradox. We also compare the achieved performance with the global optimal routing and user selfish routing, showing an inherent performance gap. These issues are important since an overlay may not be aware of the existence of other overlays in the network. The implication is that they will continually operate at this sub-optimal point and that some overlays may experience poor performance due to the unregulated application layer routing.

To bridge the performance gap or improve the fairness, two overlay pricing mechanisms are introduced to alleviate or reduce these negative effects. The pricing scheme is simple to deploy and can be implemented in a distributed fashion. Extensive simulations (both in fluid-based and packet-based) are carried out to illustrate the stability as well as the effectiveness of the distributed pricing scheme. These results not only provide some fundamental understanding of the interaction of overlays, but also point out the possibility for ISPs to perform traffic engineering so as to enhance their revenue.

10.2 Future Directions

We have witnessed the emergence of the Internet as an overlay network over the telephone systems, which triggered a fundamental shift in the structure of the telecommunications industry, with economic, policy and social implications. We believe current overlay networks on top of the Internet may signal another such shift, though the impact may not be as dramatic and far-reaching as the Internet itself.

Overlays exist for several reasons, which this thesis has tries to sort out. However, there is still a long way to go before they could be widely deployed to become a *basic* infrastructure. There are some explicit or implicit obstacles that require further attention from the researchers. Among these are:

- The lack of coordination is always an important issue. For instance, since overlays do not control the physical links themselves, routing overlays typically rely on probings of the network to measure link properties, such as available bandwidth or packet loss rate. More worrisomely, if hundreds of routing overlays coexist on the Internet, we need mechanisms to ensure that they do not overwhelm the network with redundant measurement traffic.

- An essential reason for overlays to emerge is the failure of ISPs to sell to the customers what they want to buy, which creates an opportunity for a third party to enter the market. Overlay networks are witnessed to fit into this category as a *virtual ISP*, reaping the commercial benefits of the service and leave to the ISPs only the raw business of packet carriage. The risk that the incentives of ISPs and overlay providers are not well aligned may lead to under-investment and a stagnation in innovation and upgrades.

- A more interesting concern is on the social issues. The Internet has the feature that anyone can talk to anyone freely, while overlays can be seen a cyberspace where congenial participants agree to talk among themselves, others closed out. Whether this happens and its implication to the future of the Internet, is a topic far beyond our current observation.

The Internet to some extent has matured, exhibiting a landscape of comprehensive functions and applications. Overlay networks, as a new force, represents a way to innovate and create a new order, with a set of new players, new rules and new economic relationships. We have seen the challenges and opportunities posed by overlay networks, and we believe that this trend is going to continue. By coming up with new solutions to the existing obstacles, we will witness the prosperity of overlay networks in the advent of the next generation Internet.

Bibliography

[1] J. Saltzer, D. Reed, and D. Clark, End-to-end arguments in system design, in *ACM Transactions on Computer Systems*, volume 2(4), pages 277–288, 1984.

[2] D. Anderson, H. Balakrishnan, M. F. Kaashoek, and R. Morris, Resillient overlay networks, in *Proceeding of the ACM SIGCOMM*, Karlsruhe, Germany, 2001.

[3] Gnutella, http://www.gnutella.com.

[4] L. Subramanian, I. Stoica, H. Balakrishnan, and R. Katz, Overqos: An overlay based architecture for enhancing internet qos, in *1st Symposium on Networked Systems Design and Implementation (NSDI)*, San Francisco, 2004.

[5] Y. Chu, S. Rao, S. Seshan, and H. Zhang, A case for end system multicast, in *Proceeding of ACM SIGMETRICS*, 2000.

[6] Akamai, http://www.akamai.com.

[7] I. Stoica, R. Morris, D. Karger, M. Kaashoek, and H. Balakrishnan, Chord: A scalable peer-to-peer lookup service for internet applications, in *ACM SIGCOMM 2001*, pages 149–160, San Deigo, CA, 2001.

[8] BitTorrent, http://www.bittorrent.com/.

[9] S. Savage et al., Detour: a case for informed internet routing and transport, in *IEEE Micro*, volume 19(1), pages 50–59, 1999.

[10] S. Savage, A. Collins, E. Hoffman, J. Snell, and T. Anderson, The end-to-end effects of internet path selection, in *Proceedings of ACM SIGCOMM '99*, pages 289–299, Cambridge, MA, 1999.

[11] T. Roughgarden and E. Tardos, How bad is selfish routing?, in *Journal of ACM*, volume 49(2), pages 236–259, 2002.

[12] A. Nakao, L. Peterson, and A. Bavier, A routing underlay for overlay networks, in *ACM SIGCOMM, 2003.*

[13] P. Ganesan and G. S. Manku, Optimal routing in chord., in *Proceeding of 15th Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 169–178, 2004.

[14] T. Guven, R. J. La, M. Shayman, and B. Bhattacharjee, Measurement based optimal multi-path routing, in *Proceeding of IEEE INFOCOM*, 2004.

[15] T. Guven, C. Kommareddy, R. J. La, M. A. Shayman, and B. Bhattacharjee, Measurement based optimal routing on overlay architectures for unicast sessions, in *Submitted to Computer Networks Journal on Network Modeling and Simulation*, 2004.

[16] PlanetLab, http://www.planet-lab.org/.

[17] Blumenthal, Marjory, and D. Clark, Rethinking the design of the internet: the end-to-end arguments vs. the brave new world, in *ACM Transactions on Internet Technology (TOIT)*, volume 1(1), pages 70–109, 2001.

[18] J. Jannotti, D. Gifford, K. Johnson, M. Kaashoek, and J. O'Toole, Overcast: Reliable multicasting with an overlay network, in *Proceedings of the*

4th USENIX Symposium on Operating Systems Design and Implementation (OSDI), pages 197–212, San Diego, California, 2000.

[19] I. Stoica, D. Adkins, S. Zhuang, S. Shenker, and S. Surana, Internet indirection infrastructure, in *Proceedings of ACM SIGCOOM*, 2002.

[20] D. Clark et al., The growth of internet overlay networks: Implications for architecture, industry structure and policy, in *Research Conference on Communication, Information and Internet Policy*, Vienna, 2005.

[21] Z. Duan, Z. Zhang, and Y. Hou, Service overlay networks: Slas, qos, and bandwidth provisioning, in *IEEE/ACM Transactions on Networking*, volume 11(6), pages 870–883, 2003.

[22] K. Birman, Technology challenges for virtual overlay networks, in *IEEE Transactions on Systems, Man and Cybernetics*, volume 31(4), pages 319–327, 2001.

[23] R. Johari and J. Tsitsiklis, Routing and peering in a competitive internet, in *MIT, Tech. Report*, 2003.

[24] S. Sen and J. Wang, Analyzing peer-to-peer traffic across large networks, in *IEEE/ACM Transactions on Networking*, volume 12(2), pages 219–232, 2004.

[25] G. Houston, Interconnection, peering and settlements., Available at: http://www.isoc.org/inet99/proceedings/1e/1e_1.htm, 1999.

[26] Napster, http://www.napster.org/.

[27] A. Rowstron and P. Druschel, Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems, in *Proceedings of the 18th IFIP/ACM International Conference on Distributed Systems Platforms (Middleware 2001)*, Heidelberg, Germany, 2001.

[28] B. Zhao, J. Kubiatowicz, and A. Joseph, Tapestry: An infrastructure for fault-tolerant wide-area location and routing, in *Technical Report UCB/CSD-01-1141*, University of California at Berkeley, Computer Science Department. Available at http://www.cs.berkeley.edu/~ravenen/tapestry/html/papers.html, 2001.

[29] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, A scalable content addressable network, in *Proceedings of ACM SIGCOMM'2001*, San Diego, CA, 2001.

[30] N. Harvey, M. Jones, S. Saroiu, M. Theimer, and A. Wolman, SkipNet: A scalable overlay network with practical locality properties, in *Proceedings of Fourth USENIX Symposium on Internet Technologies and Systems (USITS'03)*, 2002.

[31] R. Dingledine, N. Mathewson, and P. Syverson, Tor: The second-generation onion router, in *Proceedings of the 13th USENIX Security Symposium*, pages 303–320, 2004.

[32] *Project IRIS*, Available at http://project-iris.net/.

[33] A. Keromytis, V. Misra, and D. Rubenstein, Sos: Secure overlay services, in *Proceedings of the Sigcomm*, pages 73–86, Pittsburgh, PA, 2002.

[34] I. Clarke, O. Sandberg, B. Wiley, and T. Hong, Freenet: A distributed anonymous information storage and retrieval system, in *Proceedings of Int. Workshop Design Issues in Anonymity and Unobservability*, pages 46–66, Berkeley, CA, 2001.

[35] *Entropy (Emerging Network To Reduce Orwellian potency Yield) Project*, Available at http://entropy.stop1984.com.

[36] M. Waldman, A. Rubin, and L. Cranor, Publius: A robust, tamper-evident, censorship-resistant and source-anonymous web publishing system, in *Proceedings of the 9th USENIX Security Symposium*, pages 59–72, Pittsburgh, PA, 2000.

[37] N. Feamster, M. Balazinska, G. Harfst, H. Balakrishnan, and D. Karger, Infranet: Circumventing web censorship and surveillance, in *Proceedings of the 11th USENIX Security Symposium*, 2002.

[38] M. Waldman and D. Mazieres, Tangler: a censorship-resistant publishing system based on document entanglements, in *Proceedings of the 8th ACM Conference on Computer and Communication Security (CCS 2001)*, pages 126–135, 2001.

[39] H. Eriksson, Mbone: the multicast backbone, in *Proceedings of Communication of the ACM*, volume 37(8), pages 54–60, 1994.

[40] *Skype*, Available at http://www.skype.com.

[41] K. Fall, A Delay Tolerant Networking Architecutre for Challenged Internets, in *Proceedings of ACM SIGCOOM'03*, 2003.

[42] B. Chandra, M. Dahlin, L. Gao, and A. Nayate, End-to-end WAN service availability, in *Proceedings of 3rd USITS*, pages 97–108, San Francisco, CA, 2001.

[43] V. Paxson, End-to-end routing behavior in the internet, in *IEEE/ACM Transactoins on Networking*, volume 5(5), pages 601–615, 1997.

[44] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, Delayed internet routing convergence, in *SIGCOMM*, pages 175–187, 2000.

[45] M. Seshadri and R. H. Katz, Dynamics of simultaneous overlay network routing, UC Berkeley, CSD-03-1291, 2003.

[46] R.Keralapura, N. Taft, C.N.Chuah, and G.Iannaconne, Can ISPs Take the Heat from Overlay Networks?, in *ACM HotNets*, 2004.

[47] R. Keralapura, C. Chuah, N. Taft, and G. Iannaccone, Can coexisting overlays inadvertently step on each other?, in *13th IEEE International Conference on Network Protocols (ICNP)*, Boson, MA, 2005.

[48] Y. Liu, H. Zhang, , W. Gong, and D. Towsley, On the interaction between overlay routing and underlay routing, in *Proceeding of IEEE INFOCOM'05*, Miami, FL, USA, 2005.

[49] A. Myers, P. Dinda, and H. Zhang, Performance characteristics of mirror servers on the internet, in *Proceedings of IEEE INFOCOM'99*, New York, NY, 1999.

[50] P. Rodriguez, A. Kirpal, and E. Biersack, Parallel access for mirror sites in the Internet, in *Proceedings of IEEE INFOCOM'00*, Tel-Aviv, Israel, 2000.

[51] E. Zegura, M. Ammar, Z. Fei, and S. Bhattacharjee, Application-layer anycasting: a server selection architecture and use in a replicated web service, in *IEEE/ACM Transaction on Networking*, volume 8(4), pages 455–466, 2000.

[52] B. Krishnamurthy, C. Wills, and Y. Zhang, On the use and performance of content distribution networks, in *Proceedings of ACM Sigcomm Internet Measurement Workshop*, 2001.

[53] Herscovitz, Secure virtual private networks: the future of data communications, in *International Journal of Network Management*, volume 9(4), 1999.

[54] S. Androutsellis-Theotokis and D. Spinellis, A survey of peer-to-peer content distribution technologies, in *ACM Computing Surveys (CSUR)*, volume 36(4), 2004.

[55] R. Dingledine, *Anonymity bibliography*, Available at http://www.freehaven.net/anonbib/.

[56] S. Milberg, S. Burke, H. Smith, and E. Kallman, Values, personal information privacy, and regulatory approaches, in *Communications of the ACM*, volume 38(12), 1995.

[57] *I2P Anonymous Network*, Project website available at http://www.i2p.net/.

[58] Dabek et al., Wide-area cooperative storage with cfs, in *ACM SOSP 2001*, Banff, 2001.

[59] Rowstron, Antony, and P. Druschel, Storage management and chching in past: a large-scale, persistent peer-to-peer storage utility, in *mimoe*, 2001.

[60] M. Osborne and A. Rubinstein, *A Course in Game Theory*, The MIT Press, 1957.

[61] J. ROSEN, Existence and uniqueness of equilibrium point for concave n-person games, in *Econometrtca*, volume 3(33), pages 520–534, 1965.

[62] L. Qiu, Y. R. Yang, Y. Zhang, and S. Shenker, On selfish routing in internet-like environments, in *SIGCOMM, 2003.*

[63] H. Xie, L. Qiu, Y. R. Yang, and Y. Zhang, On self-adaptive routing in dynamic environments, in *Proceedings of International Conference on Network Protocols*, Berlin, Germany, 2004.

[64] D. Fudenberg and J. Tirole, *Game Theory*, The MIT Press, 1957.

[65] D. Bertsekas and R. Gallager, *Data Networks*, Prentice-Hall, Inc, Englewood Cliffs, NJ, 1992.

[66] I. Castineyra, N. Chiappa, and M. Steenstrup, *The Nimrod Routing Architecture, RFC, 1992*, IETF, 1996.

[67] H. Zhang, J. Kurose, and D. Towsley, Can an overlay compensate for a careless underlay, in *25th IEEE INFOCOM 2006*, Barcelona, Spain, 2006.

[68] W. Jiang, D.-M. Chiu, and J. C. S. Lui, Interaction of Overlay Networks: Properties and Implications, in *MAthematical Modelling and Analysis (MAMA) 2005 Workshop*, 2005.

[69] W. Jiang, D.-M. Chiu, and J. C. S. Lui, On the interaction of multiple overlay routing, in *IFIP WG7.3 Performance*, Juan-les-pins, France, 2005.

Appendix A

Proof of Existence of Nash Equilibrium

Lemma 3. *The preference relation $\succsim_s$ in an overlay optimal routing game $G_{\text{overlay}}\langle \mathcal{N}, (\Gamma_s), (\succsim_s)\rangle$ is continuous and quasi-concave on Γ_s.*

Proof: It is equivalent to prove that the payoff function for each player s is continuous and quasi-concave on Γ_s. Since the payoff function for each overlay s is the negative of average delay, thus it is equivalent to prove that function $\text{delay}^{(s)}(y^{(s)}; y^{(-s)})$ is continuous and quasi-convex on Γ_s. The continuity of $\text{delay}^{(s)}(y^{(s)}; y^{(-s)})$ is trivial if we require the delay function of each link $j \in \mathcal{J}$ to be continuous. To prove quasi-convexity of $\text{delay}^{(s)}(y^{(s)}; y^{(-s)})$ on Γ_s, it suffices to show that

$$z^T \nabla \text{delay}^{(s)}(y^{(s)}) = 0 \;\Rightarrow\; z^T \nabla^2 \text{delay}^{(s)}(y^{(s)}) z > 0 \tag{A.1}$$

for all $y^{(s)} \in \Gamma_s$ and for all $z \in \mathbb{R}^{r_s}, z \neq 0$. We have

$$\text{delay}^{(s)}(y^{(s)}) = \frac{1}{\sum_{f \in \mathcal{F}_s} x_f} \cdot y^{(s)^T} \Big[A^{(s)^T} \mathcal{D}\Big(\sum_i A^{(i)} y^{(i)} \Big) \Big]. \tag{A.2}$$

We can write out the gradient vector and the Hessian matrix of the delay function for overlay s as:

$$\nabla \text{delay}^{(s)} = A^{(s)^T} \mathcal{D}\Big(\sum_i A^{(i)} y^{(i)} \Big) + A^{(s)^T} \nabla \mathcal{D}\Big(\sum_i A^{(i)} y^{(i)} \Big) A^{(s)} y^{(s)} \tag{A.3}$$

and

$$\nabla^2 \text{delay}^{(s)} = 2A^{(s)\mathcal{T}} \nabla \mathcal{D}\Big(\sum_i A^{(i)} y^{(i)}\Big) A^{(s)} + \Big[A^{(s)\mathcal{T}} \nabla^2 \mathcal{D}\Big(\sum_i A^{(i)} y^{(i)}\Big) A^{(s)}\Big] A^{(s)} y^{(s)}. \tag{A.4}$$

As defined in the previous section, $A^{(s)}$ is the sub-routing matrix for overlay s. We denote $a_{kq}^{(s)}$ as the k-row q-column element of $A^{(s)}$. $\nabla \mathcal{D}$ is an $m \times m$ diagonal matrix and $\nabla \mathcal{D}\Big(\sum_i A^{(i)} y^{(i)}\Big) = diag\big\{d_1'(l_1), \ldots, d_m'(l_m)\big\}$, while $\nabla^2 \mathcal{D}$ is a $1 \times m$ vector of which each element is an $m \times m$ diagonal matrix, where $\nabla^2 \mathcal{D}_k = diag\big\{d_k''(l_k),\ 0, \ldots,\ 0\big\}$.

By simple algebraic manipulation, we rewrite $\nabla^2 delay^{(s)}$ as an r_s by r_s matrix wherein $r_s = \sum_{f \in \mathcal{F}_s} |\mathcal{R}_f|$. We denote h_{pq} as the p-row q-column element of the Hessian matrix. Also we denote $a_{pq}^{(s)}$ as the p-row q-column element of $A^{(s)}$. Then we have the following:

$$h_{pq} = 2 \sum_{k=1}^{m} a_{kp}^{(s)} a_{kq}^{(s)} d_k'(l_k) + \sum_{k=1}^{m} g_k^{(s)} a_{kp}^{(s)} a_{kq}^{(s)} d_k''(l_k) \tag{A.5}$$

where $g_k^{(s)} = \sum_{r=1}^{r_s} a_{kr}^{(s)} y_r^{(s)}, r \in \mathcal{R}_f, f \in \mathcal{F}_s$. To prove the quasi-convexity of delay$^{(s)}$, denote $z = (z_1, z_2, \ldots, z_{r_s})^{\mathcal{T}}$ as a non-zero vector.

$$\begin{aligned} z^{\mathcal{T}} \nabla^2 \text{delay}^{(s)} z &= \sum_{p,q=1}^{r_s} \sum_{k=1}^{m} \Big(2a_{kp}^{(s)} a_{kq}^{(s)} d_k'(l_k) + g_k^{(s)} a_{kp}^{(s)} a_{kq}^{(s)} d_k''(l_k)\Big) z_p z_q \\ &= \sum_{k=1}^{m} \Big[\sum_{p,q=1}^{r_s} a_{kp}^{(s)} a_{kq}^{(s)} \Big(2d_k'(l_k) + g_k^{(s)} d_k''(l_k)\Big) z_p z_q\Big]. \end{aligned} \tag{A.6}$$

Let us denote $\beta_k = 2d_k'(l_k) + g_k^{(s)} d_k''(l_k)$, then

$$\begin{aligned} z^{\mathcal{T}} \nabla^2 \text{delay}^{(s)} z &= \sum_{k=1}^{m} \beta_k \Big(\sum_{p,q=1}^{r_s} a_{kp}^{(s)} a_{kq}^{(s)} z_p z_q\Big) \\ &= \sum_{k=1}^{m} \beta_k \Big(\sum_{r=1}^{r_s} a_{kr}^{(s)} z_r\Big)^2 \end{aligned} \tag{A.7}$$

Since $d_k'(l_k) > 0, d_k''(l_k) > 0$ and $g_k^{(s)} \geq 0$, we have $\beta_k > 0$, $\big(\sum_{r=1}^{r_s} a_{kr}^{(s)} z_r\big)^2 \geq 0$. Furthermore, if the rank of matrix $A^{(s)}$, *e.g.* rank$(A^{(s)}) \geq r_s$, we have

$\left(\sum_{r=1}^{r_s} a_{kr}^{(s)} z_r\right)^2 > 0$ for some $k \in \{1, \ldots, m\}$ and all non-zero vector z. Based on the above derivations we have $z^T \nabla^2 \text{delay}^{(s)} z > 0$. Finally we have proved that $\nabla^2 \text{delay}^{(s)}$ is convex on $y^{(s)}$, and of course quasi-convex on $y^{(s)}$. This completes the proof that the preference relation $\succsim_i$ in the overlay optimal routing game G_{overlay} is continuous and quasi-concave. ∎