공학박사학위논문

Resource Allocation, Pricing, and Failure Management of Virtual Networks

가상 네트워크의 자원 할당, 가격 결정 및 고장 관리

2013년 8월

서울대학교 대학원

전기•컴퓨터공학부

이 승 호

Resource Allocation, Pricing, and Failure Management of Virtual Networks

가상 네트워크의 자원 할당, 가격 결정 및 고장 탐지

지도교수 서승우

이 논문을 공학박사 학위논문으로 제출함
2013년 8월

서울대학교 대학원
전기·컴퓨터공학부
이승호

이승호의 공학박사 학위논문을 인준함
2012년 12월

위원장 박세웅
부위원장 서승우
위원 권태경
위원 박성용
위원 정한유

Abstract

Network virtualization is an emerging technology that enables the dynamic partitioning of a shared physical network infrastructure into multiple virtual networks. Because of its flexibility in resource allocation and independency among virtual networks, the network virtualization technology has not only been mainly deployed to build a testbed network, but also has come to be regarded as a cost-effective solution for diversifying the Internet. As a means of building the multi-layered Internet, network virtualization still faces a number of challenging issues that need to be addressed. This dissertation deals with several important research topics and provides effective solutions in network virtualization environment.

First, I focus on the optimal partitioning of finite substrate resources for satisfying the diverse QoS requirements of virtual networks. I formulate virtual network partitioning problem as a mixed integer multi-commodity flow problem. Then, to tackle the structural complexity of the problem, I propose a simple heuristic based on shortest path routing algorithm. By conducting large-scale network experiments, I verify the efficiency and scalability of the heuristic.

Next, I propose an economic model for tiered access service in virtual networks in order to remedy the deficiency of the existing tiered service schemes. I first derive a sufficient condition for stability of user subscription dynamics, and find the optimal pricing and capacity partitioning by addressing the revenue maximization problem of the tiered access service in a network virtualization environment. Numerical results show that the tiered service can be more profitable than the non-tiered service under proper pricing and capacity partitioning conditions.

Last, I develop a fast and effective failure recovery mechanism through inter-virtual network traffic switching in virtual networks. The proposed failure recovery mechanism neither has topological constraints for the existence of backup paths, nor requires the pre-computation of them, but nevertheless guarantees as fast recovery as the existing failure recovery methods.

This dissertation aims to address important issues in the virtual network-based Internet. I believe that the analysis and results in this dissertation will provide useful guidelines to improve the Internet.

Keywords: network virtualization, service differentiation, pricing, tiered access service, capacity partitioning, failure recovery

Student number: 2007-21044

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and Motivation

Network virtualization is an emerging technology for diversifying the Internet. Network virtualization allows multiple virtual networks to coexist on a shared physical infrastructure. Each virtual network can be customized for a specific service or user group. Because of the flexibility in resource allocation and independency between different virtual networks, it has been mainly deployed to build a testbed network such as PlanetLab [1] and GENI [2]. However, since several pluralist studies suggested that virtualization should become a fundamental attribute of the Internet [3–5], network virtualization has also come to be regarded as a means of diversifying the Internet, not just a tool for experimentation. That is, since a network virtualization environment accommodates different types of services within logically-partitioned virtual networks simultaneously, the network virtualization environment itself can serve as a *layered* network architecture for offering differentiated services.

Over the last decade, a number of studies have been devoted to the technical and theoretical issues related to network virtualization. Among them, the majority of early studies [6–9] and even several recent studies [10–13] have focused on the on-demand embedding of virtual resources (i.e., virtual nodes and links) onto the underlying physical resources since it is of practical importance in operating and managing the testbed network. As a means of building the *layered* Internet, however, network virtualization still faces a number of challenging issues that need to be addressed, including admission control, addressing, failure handling, security and privacy, and network economics [14].

In this dissertation, I raise several important research issues in network virtualization environment and provide effective solutions for them.

1.2 Contributions and Outline of the Dissertation

The main contributions of this dissertation are summarized as follows:

- I propose an efficient and effective resource allocation method to offer differentiated service in virtual networks.
- I develop an economic model to analyze the user subscription behavior and the ISP revenue of tiered access service in virtual networks.
- I propose a fast failure recovery method based on inter-virtual network traffic switching in virtual networks.

This dissertation aims to address new and important issues in the virtual network-based Internet. The theoretical analysis and outstanding results in this dissertation are expected to provide useful guidelines to improve the current Internet and to design the Future Internet architecture.

The rest of this dissertation is outlined as follows:

In Chapter 2, I focus on the optimal resource allocation to offer service level differentiation in virtual networks. In a network virtualization environment, since multiple virtual networks share the same underlying physical resources, of utmost importance is efficient and equitable partitioning of the substrate network. To this end, I first formulate the substrate partitioning problem as a mixed integer multi-commodity flow problem which involves the capacity and service level constraints for multi-layered virtual networks. Then, to tackle the structural complexity of the problem, I propose a simple heuristic based on the shortest path routing algorithm. Lastly, I conduct a large-scale experiment to verify the accuracy and efficiency of the proposed heuristic.

In Chapter 3, I propose a network architectural solution that utilizes network virtualization technology as a means of providing tiered access service in order to remedy the deficiency of the existing tiered service models and to overcome the inherent limitation of the current Internet architecture. I first develop an economic model for the tiered access service based on virtual networks, which accounts for the user subscription dynamics taking into account the impacts of access rate regulation. Then, I find the optimal pricing and capacity partitioning by addressing the revenue maximization problem in the tiered service.

In Chapter 4, I develop a fast and effective failure recovery mechanism through inter-virtual network traffic switching in virtual networks. Throughout the Chapter, I carefully design the details of the inter-virtual network traffic switching mechanism and failure recovery process. Since I acknowledge that the proposed failure recovery mechanism has a potential risk that the switched traffic could cause QoS degradation in the neighbor virtual network, I pay careful attention to the analysis of this possible adverse effect and the design of a practical traffic switching scheme that can minimize the QoS degradation.

Finally, Chapter 5 summarizes the contributions and conclude the dissertation.

Chapter 2

Effective Partitioning for Service Level Differentiation in Virtual Networks

2.1 Introduction

With the advent of heterogeneous applications, there has been a growing need for service differentiation in the Internet. However, it is well-known that the simple best-effort service model of the current IP network does not provide QoS guarantee for specific applications (e.g., delay guarantee for VoIP and video streaming). Over the last two decades, a large number of studies have attempted to develop an effective QoS framework which can be accommodated within the current Internet architecture [15–19], but due to the lack of scalability and implementability, none of them have been successfully deployed. In this respect, QoS is regarded as one of the most important challenging issues for designing Future Internet architecture [20].

In the meantime, network virtualization has emerged as a key technology for diversifying the Internet. Network virtualization allows multiple virtual

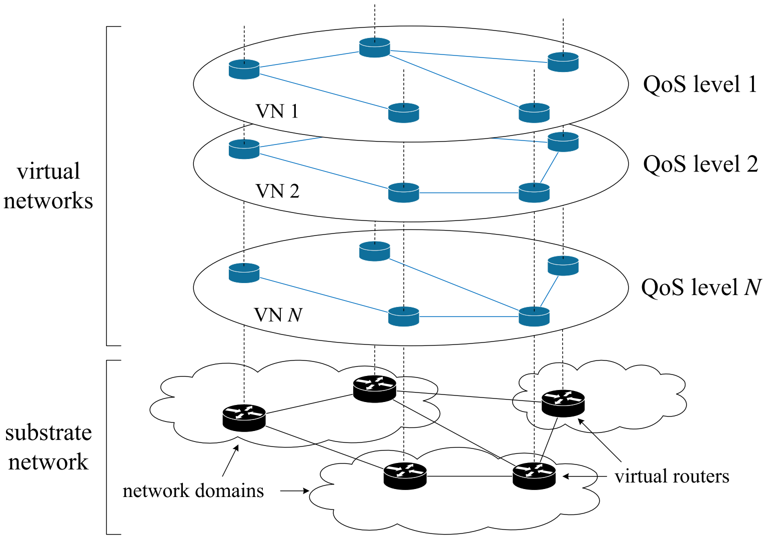


Figure 2.1: Network virtualization environment for offering differentiated services.

networks (VNs) to coexist on a shared physical infrastructure. Each VN can be customized for a specific service or user group. Accordingly, network virtualization has been widely deployed for building testbed networks such as PlanetLab [1], VINI [21], and GENI [2]. Since the VN-based testbed provides each user with a dedicated wide-ranging experimental network, users can run and evaluate their applications or protocols without any interference from the others.

Since a network virtualization environment (NVE) accommodates different types of services within logically-partitioned VNs simultaneously, the NVE itself can serve as a layered network architecture for offering differentiated services[1] as described in Figure 2.1. Its architectural features can simplify the core functions for service differentiation, including service level agreement,

---

[1] The term *differentiated services* is also used to describe a QoS mechanism *DiffServ*. We use this term in a general sense, without referring to a specific mechanism or protocol.

network management, configuration, billing, and customer support. Besides, in the NVE, since the physical resources such as router CPU and link bandwidth can be flexibly assigned to each VN, a certain degree of QoS guarantee for each VN is achievable with an effective resource allocation. Thus, the network virtualization can be a good alternative in the sense of service differentiation, which enables to overcome the inherent limitations of the single network architecture.

In this Chapter, we study the design of NVE to support differentiated services.[2] Specifically, we focus on the effective partitioning of substrate resources in order for every VN to satisfy its service level requirement. To this end, we first formulate the substrate partitioning problem taking into account a stochastic traffic demand model and QoS constraints. The problem finds the optimal path for every traffic flow and the optimal link capacity along the path. Due to the complex dependency among variables, however, the problem is computationally intractable. Thus, to solve the problem, we provide a simple heuristic based on the shortest path routing. The proposed heuristic presents a reasonable approximation to the global optimum and shows outstanding performance for differentiated service level guarantee in comparison with the well-known shortest path routing method. Finally, we verify the efficiency and scalability of the heuristic by conducting extensive simulations for various traffic demands.

The main contributions of this Chapter can be summarized as follows: First, we present a theoretical framework for the VN-based service differentiation. Since the service partitioning problem needs to cover some additional

---

[2] This work was conducted in collaboration with Seung-Ho Yoo, Moon-Young Jeong, and Prof. Seung-Woo Seo, so the pronoun *we* is used throughout the Chapter.

assumptions and constraints in comparison with the existing problems, it is far more difficult to address the problem formally. By defining a practical traffic demand model and reasonable QoS metric, we formulate the substrate partitioning problem as a mixed integer multi-commodity flow problem. Next, we provide a different approach to the resource allocation problem for VNs. Most of the existing studies have dealt with the on-demand embedding of a VN request, which would benefit the resource allocation for the VN-based testbed network. However, we consider the (initial) planning of NVE as a layered network architecture. This approach is expected to provide a useful foundation for designing VN-based Future Internet architecture.

The remainder of this Chapter is organized as follows. Section 3.7 discusses related studies on resource allocation for VN. Section 2.3 describes the underlying network model and assumptions on traffic demand and QoS metric. In section 2.4, we formulate the substrate partitioning problem, and simplify the problem by decomposing. In section 2.5, we provide a heuristic based on the shortest path routing. Section 2.6 presents the experimental results that evaluate the performance of the proposed heuristic, and section 2.7 summarizes the Chapter.

2.2 Related Work

Network virtualization has been introduced as a means of overcoming the architectural limitations of the Internet in [3–5]. These studies emphasize the necessity of diversifying the Internet using VNs. Thereafter, a large number of studies have been conducted on the resource allocation problem for VNs. They mainly focus on how to effectively and efficiently map a VN request onto the substrate network [6–11, 22–24]. For a given VN request containing a set of virtual nodes and links that have fixed amounts of capacity requirements, this VN embedding problem aims to find a set of paths which can be accommodated within the remaining resources of the substrate network. Since the substrate resources are limited, and some of them may already be exhausted by the previous requests, the problem should be able to consider admission control for additional unacceptable requests. The VN embedding problem is well-suited to modeling the VN assignment in VN-based testbed. However, since the on-demand embedding for each VN request cannot balance the QoS requirements of different VNs, it is inapplicable to designing the NVE for offering differentiated services.

Meanwhile, [25] have proposed service overlay network to achieve the end-to-end QoS guarantee for one or more value-added Internet services on a dedicated (overlay) network. In the service overlay network, since the end-to-end path for each traffic demand is predetermined, the only concern is to find out the optimal capacity for an overlay network on all the links along the path. This capacity provisioning problem takes into account the end-to-end service QoS. Insufficient provisioning can impose a penalty on the service provider due to the violation of the service QoS, while over-provisioning increases the cost for leasing resources. Under this trade-off, the problem aims to maximize the net income of the service provider. The capacity provisioning problem, however, considers the optimal capacity provisioning only for a single overlay network without capacity constraints of the substrate network. Therefore, it is also difficult to directly apply this framework to the substrate partitioning problem.

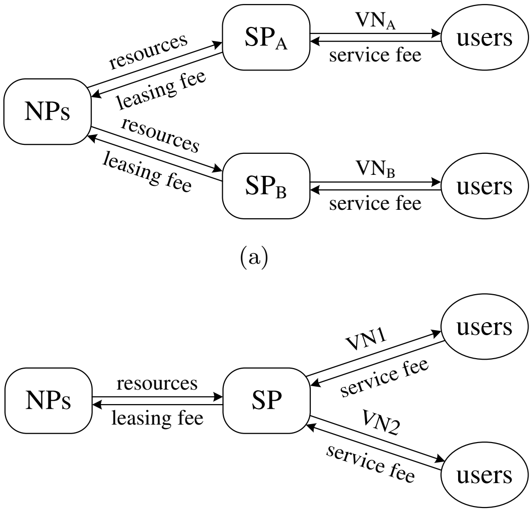
Here, since we need to be able to consider both the effective embedding of traffic demands and the optimal capacity provisioning for VNs together, it is more difficult to identify and formulate the problem. In the next section, we define the underlying assumptions and models required to formulate the substrate partitioning problem.

2.3 Model and Assumption

2.3.1 Business Model

We first describe the business relationship for VNs. In a general NVE, multiple service providers generate VNs to offer end-to-end customized services for users by reserving network resources from one or more network providers. Each service provider pays some cost for leasing a certain amount of resources from network providers, and charges users for the service. Accordingly, from a perspective of service providers, how to minimize the physical resources for creating a VN is of utmost importance in order to increase the net income. In contrast, network providers are interested in maximizing the network utilization. Figure 2.2(a) shows the business relationship in the general NVE.

Contrarily, in the NVE offering differentiated services, a service provider generates multiple VNs to offer differentiated services, as shown in Figure 2.2(b). The service provider leases a fixed amount of physical resources from one or more network providers, and appropriately partition the resources taking into account the amount of traffic demands for each VN. Note here that the net income of the service provider is fixed, since both the amount of total physical resources and the number of users for each VNs are invariant over time. The service provider has no motivation for optimal partitioning. In



(a)

(b)

Figure 2.2: Business relationship in (a) a general network virtualization environment, and (b) the network virtualization environment offering differentiated services.

order to resolve this problem, we assume that inappropriate partitioning of VNs would potentially impose a loss on the service provider. A continuous violation of QoS guarantee due to insufficient resources for a VN can eventually decrease the number of users of that VN, which leads to a decrease in the net income. This assumption is expected to motivate the service provider to partition VNs efficiently.

2.3.2 Network Model

We consider a NVE in which N VNs share a substrate network. We assume that the number of VNs N is determined according to the service policy of service provider, and does not change over time unless there is an additional need for new VN.

We assume that VNs are completely isolated from each other. That is, the traffic on a VN cannot be switched to another VN, and the congestion occurring in a VN does not affect the network performance of the other VNs.

We model the substrate network as a directed graph $G = (V, E)$, where V is the set of nodes and E is the set of links. Each link $l_{ij} \in E$ has a limited bandwidth capacity C_{ij}, which can be shared among a couple of VNs. For example, if some traffic on the nth VN is routed through the link l_{ij}, the VN n has bandwidth capacity c_{ij}^n on the link. For simplicity, we assume that nodes have infinite capacity.

2.3.3 Traffic Demands

Traffic demands from each end node vary over time. Thus, the amount of traffic loads aggregated at each link is unpredictable. In order to provision an adequate amount of bandwidth capacity for each virtual link to accommodate the varying traffic loads, and eventually satisfy the QoS requirement for each VN, we need to capture the variation of traffic loads. However, it is almost impossible to accurately measure traffic loads for every link in real time.

Accordingly, we employ a stochastic traffic demand model. Let $\rho^n(s, d)$ denote the traffic demand from a source node s to a destination node d on the nth VN. We assume that the traffic demand vector ρ^n varies according to some distribution with the average value of $\bar{\rho}^n$, which can be measured by service provider. The average traffic demand $\bar{\rho}^n$ is assumed to be invariant during the time period T, which is the basic time unit for the partitioning of substrate resources. We assume that T is determined by taking into account the service usage pattern of end users (i.e., business and non-business hours).

We assume that the short-term fluctuation of traffic demands follows $M/G/\infty$ traffic model. Since the $M/G/\infty$ queue is known to generate the asymptotically self-similar traffic, $M/G/\infty$ traffic model has been regarded as the most realistic traffic model for the Internet in numerous studies [26–29]. Consider an $M/G/\infty$ queue, where the service time has heavy-tailed distribution with a finite mean. Let X_t denote the number of customers in the system at time t, then X_t has a Poisson marginal distribution with mean $\lambda\mu$ [30], where λ is the arrival rate to the system and μ is the mean service time. In the $M/G/\infty$ traffic model, the counting process X_t is mapped into the traffic demands on each path with the average value of $\bar{\rho}_r$. As the traffic demands for all end-to-end paths are assumed to be independent from each other, the amount of traffic loads aggregated at each link also has the Poisson marginal distribution. Then the average traffic loads on link (i, j) is given by

$$\bar{\rho}_{ij} = \sum_{r:(i,j)\in r} \bar{\rho}_r .$$

2.3.4 QoS Metric

We assume that every VN has its own QoS requirement. In order to quantify the end-to-end QoS for each traffic demand, we need a QoS metric. QoS is generally identified by diverse performance measures such as throughput, delay, jitter, loss rate, and so on. None of these performance measures, however, can be used as the unified QoS metric in our NVE, where diverse types of services are aggregated in each VN.

In this Chapter, we employ congestion probability as a QoS metric. The congestion probability for single link is defined as the probability that the traffic loads on the link exceed a certain level of link utilization. Since the congestion probability is given as the function of bandwidth capacity and traffic loads, it is easily applicable to diverse services. Let K_{ij} denote the congestion probability for link (i, j), then $K_{ij} = \Pr\{\rho_{ij} > \eta C_{ij}\}$, where η is the link utilization threshold. For $M/G/\infty$ traffic model, the congestion probability for link (i, j) is given by

$$K_{ij}(\bar{\rho}_{ij}, C_{ij}) = \sum_{k=\eta C_{ij}+1}^{\infty} \frac{\bar{\rho}_{ij}{}^{k}}{k!} e^{-\bar{\rho}_{ij}}.$$

Since the $M/G/\infty$ traffic model is defined in a discrete space, $K_{ij}(\bar{\rho}_{ij}, C_{ij})$ is valid only for integer value of ηC_{ij}. For non-integer ηC_{ij}, we can approximate K_{ij} by using linear interpolation.[3]

Similarly, we can derive the link congestion probability for each VN. Recall that VNs are assumed to be completely isolated from each other. The traffic on a VN does not affect the performance of other VNs. The traffic loads for nth VN on link (i, j) are, therefore, bounded by the virtually partitioned bandwidth capacity c_{ij}^{n}. Then the link congestion probability for the nth VN is given by

$$K_{ij}(\bar{\rho}_{ij}^{n}, c_{ij}^{n}) = \sum_{k=\lfloor \eta c_{ij}^{n} \rfloor+1}^{\infty} \frac{\bar{\rho}_{ij}^{n\,k}}{k!} e^{-\bar{\rho}_{ij}^{n}}.$$

Now we consider the end-to-end congestion probability. Unfortunately, we cannot in general obtain the multi-hop congestion probability even if all of the link congestion probabilities along the path are given, since the congestion events occurring on adjacent links are likely correlated. To overcome

---

[3] For non-integer ηC_{ij}, let K_{ij}^{-} and K_{ij}^{+} denote the link congestion probability for $\lfloor \eta C_{ij} \rfloor$ and $\lfloor \eta C_{ij} \rfloor + 1$, respectively. Then,

$$K_{ij} = K_{ij}^{-} + (\eta C_{ij} - \lfloor \eta C_{ij} \rfloor)(K_{ij}^{+} - K_{ij}^{-}).$$

this difficulty, many other studies on traffic analysis and capacity planning in the telephone networks assume that the correlation among adjacent links is negligibly small, i.e. the congestion events occurring on different links are independent [25, 31]. In this Chapter, for ease of analysis, we also adopt this link independency assumption. Then we can express the end-to-end congestion probability in terms of the link congestion probability. The end-to-end congestion probability K_r, which is the probability that at least one of the links along the path r is congested, is given by

$$K_r = 1 - \prod_{(i,j)\in r} \left(1 - K_{ij}\right).$$

Generally, the link congestion probability has a very small value (i.e. $K_{ij} \ll 1$). For small K_{ij}, the following approximation holds:

$$K_r \simeq \sum_{(i,j)\in r} K_{ij}.$$

In fact, we can also consider the NVE where each VN employs a different type of QoS metric, instead of the case where VNs are differentiated according to different level for the same QoS metric. For example, a VN which is customized for VoIP can adopt the end-to-end delay as a QoS metric, while another VN for data transfer can use the loss rate. The substrate partitioning problem, which is discussed in the next section, should provide a universal framework to support different types of QoS metrics as well.

2.4 Formulation

2.4.1 Objective

We first clarify the objective of the problem. Recall that the substrate partitioning problem is required to provide differentiated QoS guarantee for each VN. To satisfy this requirement, we can first consider the optimization problem which uses the total congestion probability as the objective function, and imposes a QoS requirement constraint. Since the congestion probability of traffic demands is strongly restricted, the solution of the problem will obviously satisfy the above requirement for normal traffic. However, if some traffic demands unavoidably violate their QoS requirements with an increase of the overall traffic volume, the problem will have no feasible solution. Even in that situation, we should be able to provide an optimal solution which minimizes the total congestion probability.

Alternatively, we can merge the QoS requirement constraint into the objective function. That is, we can consider an objective function which includes the terms for congestion probability requirement as well as the total congestion probability. To support this, we newly define a congestion externality, which means the dissatisfaction degree of users for congestion. Let Q denote the congestion externality for each traffic demand. Then Q is given by

$$Q = \left(\frac{K_r}{\varepsilon^n} \right)^{\varphi}$$

where ε^n is the QoS requirement of VN n, and φ is the dissatisfaction index, which is given as $\varphi > 1$. Since the congestion externality Q normalizes the end-to-end congestion probability K_r by dividing it by the QoS requirement ε^n, fairness between VNs which have different QoS requirement is achieved.

Since the dissatisfaction index φ has a value greater than 1, the dissatisfaction degree of users increases exponentially as K_r increases, which means that minimizing the total congestion externality for all traffic demands leads to minimizing the total QoS violation. That is, minimizing the total congestion externality guarantees both efficiency and fairness for offering differentiated services. We employ the total congestion externality as the objective function of the substrate partitioning problem.

2.4.2 Substrate Partitioning Problem

We formulate the substrate partitioning problem as a mixed integer multi-commodity flow problem. We consider each traffic demand as a commodity. Let $m^n \in M^n$ denote the mth commodity of the nth VN. Then we can represent the average traffic demand for each commodity as $\bar{\rho}^n(m^n)$.

We present the mixed integer multi-commodity flow problem as the following manner:

Problem 0 *Substrate Partitioning*

Variables:

- $x_{ij}^n(m^n)$: A binary decision variable, which equals to 1 if a commodity m of VN n is routed on the link (i,j) and 0 otherwise.
- c_{ij}^n: A capacity variable, which represents the amount of bandwidth capacity assigned to VN n at the link (i,j).

Objective:

$$\text{minimize} \qquad \sum_{n} \sum_{m^n} \left(\frac{\sum\limits_{(i,j)\in P(m^n)} K_{ij}(y_{ij}^n, c_{ij}^n)}{\varepsilon^n} \right)^{\varphi} \tag{2.1}$$

Constraints:

$$\sum_j x_{ij}^n(m^n) - \sum_j x_{ji}^n(m^n) = \begin{cases} 1 & \text{if } i = \text{source} \\ -1 & \text{if } i = \text{destination} \\ 0 & \text{otherwise} \end{cases}, \ \forall i \in V,\ m^n \in M^n,\ n \in N \tag{2.2}$$

$$x_{ij}^n(m^n) = \begin{cases} 1 & \text{if } (i,j) \in P(m^n) \\ 0 & \text{otherwise} \end{cases}, \quad \forall (i,j) \in E,\ m^n \in M^n,\ n \in N \tag{2.3}$$

$$y_{ij}^n = \sum_{m^n} \bar{\rho}^n(m^n) x_{ij}^n(m^n), \quad \forall (i,j) \in E,\ n \in N \tag{2.4}$$

$$\sum_n y_{ij}^n \le C_{ij}, \quad \forall (i,j) \in E \tag{2.5}$$

$$\sum_n c_{ij}^n = C_{ij}, \quad \forall (i,j) \in E \tag{2.6}$$

Remarks:

- Objective function (2.1) aims to minimize the total congestion externality. ε^n is the QoS requirement of VN n, and φ is the dissatisfaction index. $P(m^n)$ represents the end-to-end path for the mth commodity of VN n.

- Constraints (2.2) and (2.3) refer to the nodal flow conservation, which state that the net flow to a node is zero, except for the source and the destination node.

- In constraint (2.4), y_{ij}^n denotes the average traffic loads for the nth VN at link (i, j).

- Constraints (2.5) and (2.6) represent capacity constraints. At each link, the sum of the traffic loads for each VN is bounded by the physical bandwidth capacity C_{ij}, and the sum of the bandwidth capacity for each VN is equal to C_{ij}.

2.4.3 Decomposition

The above substrate partitioning problem has two variables to optimize, $x_{ij}^n(m^n)$ and c_{ij}^n. These are dependent with each other. The decision variable $x_{ij}^n(m^n)$ determines the end-to-end path for every traffic demand. Then we can obtain the total average traffic loads for each VN at each link, y_{ij}^n. The link congestion probability K_{ij}^n is given as the function of y_{ij}^n and the capacity variable c_{ij}^n. We can find the optimal value of c_{ij}^n which minimizes the total congestion externality for every traffic demand. This complex dependency between variables makes the problem computationally intractable.

In order to simplify the problem, we decompose the substrate partitioning problem into two sequential subproblems as follows:

Problem 1 *Feasible Path Planning*[4]

Objective:

$$\text{minimize} \qquad \textit{const.}$$

Constraints:

$$\sum_j x_{ij}^n(m^n) - \sum_j x_{ji}^n(m^n) = \begin{cases} 1 & \text{if } i = \text{source} \\ -1 & \text{if } i = \text{destination} \\ 0 & \text{otherwise} \end{cases}, \ \forall i \in V,\, m^n \in M^n,\, n \in N$$

$$x_{ij}^n(m^n) = \begin{cases} 1 & \text{if } (i,j) \in P(m^n) \\ 0 & \text{otherwise} \end{cases}, \quad \forall (i,j) \in E,\, m^n \in M^n,\, n \in N$$

$$y_{ij}^n = \sum_{m^n} \bar{\rho}^n(m^n) x_{ij}^n(m^n), \quad \forall (i,j) \in E,\, n \in N$$

---

[4] We use the term *path planning* in the sense of pre-assigning a fixed end-to-end path for each traffic demand, which is distinguished from *routing*. However, since both of them can be represented as the same optimization problem, sometimes, these terms are used interchangeably in this Chapter.

$$\sum_{n} y_{ij}^{n} \leq C_{ij}, \quad \forall (i,j) \in E$$

Problem 2 *Capacity Partitioning*

Objective:

$$\text{minimize} \qquad \sum_{n} \sum_{m^n} \left(\frac{\sum\limits_{(i,j) \in P(m^n)} K_{ij}(y_{ij}^{n}, c_{ij}^{n})}{\varepsilon^{n}} \right)^{\varphi}$$

Constraints:

$$\sum_{n} c_{ij}^{n} = C_{ij}, \quad \forall (i,j) \in E$$

Problem 1 is formulated as a feasibility problem, where the objective function is a constant (e.g. zero). The variable of the problem is a binary decision variable $x_{ij}^{n}(m^n)$, which determines end-to-end paths for all commodities under the capacity constraint. Problem 1, therefore, finds all possible combinations of end-to-end paths satisfying the constraints.

Problem 2 takes a relatively simple form, which has only a capacity constraint. For given traffic loads y_{ij}^{n}, the problem determines the optimal capacity partitioning for VNs at each link. Since the congestion probability K_{ij}^{n} is convex for $y_{ij}^{n} < \eta c_{ij}^{n}$, the objective function, which is the sum of convex functions, is also convex. Therefore, we can easily solve the problem using a numerical method such as *Newton's method*.[5]

By solving two subproblems in turn, we can obtain both the optimal end-to-end paths for traffic demands and link capacities for VNs, which minimize

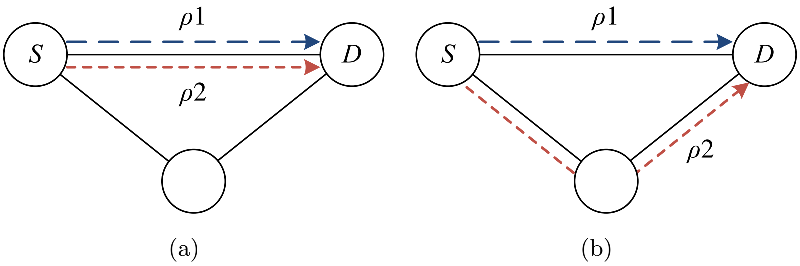
---

[5] In order to use *Newton's method*, we need to compute the first and second derivatives of K_{ij}^{n} with respect to c_{ij}^{n}. Since K_{ij}^{n} is defined only for the integer values of c_{ij}^{n}, we take the average of left and right derivatives as the value of the first derivative at each integer point. Likewise, we obtain the second derivative of K_{ij}^{n}.

the total congestion externality. This naïve approach, however, is neither practical nor scalable. First, Problem 1 is computationally intractable. There are a few heuristics and approximation algorithms for solving the integer multi-commodity flow problem, but as far as we know, none of them can be applied to the form of the feasibility problem which has a constant objective function. Next, the absence of a clear objective function in Problem 1 gives rise to too many solutions. Although we can solve Problem 2 with *Newton's method* in a few seconds, it is very cumbersome to compute the optimal capacities for all feasible path sets one by one. Thus, in order to reach the global optimum of the substrate partitioning problem quickly or achieve at least a good approximation of it, we need to be able to cope with these two obstacles.

2.5 Heuristic

In this section, we discuss how to solve the substrate partitioning problem effectively. An ideal approach for shortening the overall solving process of substrate partitioning problem is to solve every subproblem just once. That is, if we can obtain the globally optimal path set which minimizes the total congestion externality in advance to solve the capacity partitioning problem, we can avoid unnecessary repetition. However, it is impracticable to define an appropriate objective function which derives the globally optimal path set from all solutions of feasible path planning problem. Since the optimality of path set depends upon various factors such as topology and link capacities of substrate network, traffic demand distribution, and QoS requirement of each VN, we cannot easily identify the key determinants of the globally optimal path set.



(a) (b)

Figure 2.3: Path planning examples, where two traffic demands take (a) the same path, and (b) different paths.

Thus, in this study, we consider a heuristic approach to simplify the complexity of the problem. Our heuristic is based on the reformulation of the path planning problem. We redefine the objective function in the path planning problem so as to find a path set which minimizes the sum of the end-to-end congestion probabilities for all traffic demands. Then we calculate the optimal capacity partitioning for this efficiency-oriented path set in order to ensure the fairness of VNs. This approach, which basically assigns different objectives (i.e. efficiency and fairness) to each sub-problem, is expected to provide a good approximation to the optimal solution under any conditions.

Now let us investigate the optimal path planning which minimizes the total end-to-end congestion probabilities for all traffic demands. Recall that the end-to-end congestion probability for a single traffic demand approximates to the sum of congestion probabilities for all links along the path. Then, in general, shorter path is more likely to have lower congestion probability. In the case of multiple traffic demands, however, the set of shortest paths does not always minimize the total congestion probabilities. Let us consider simple examples as shown in Figure 2.3. In Figure 2.3(a), both traffic demands $\rho 1$ and $\rho 2$, which have the same source and destination nodes, are routed along the same (shortest) path, but in Figure 2.3(b), they take different paths. In terms of the total end-to-end congestion probabilities, we cannot conclude that one is better than another. Since the link congestion probability depends exponentially on the average traffic loads, if the amount of each traffic demand is sufficiently large, the effect of merging traffic loads of different links on the total end-to-end congestion probabilities can overwhelm the effect of shorter paths. That is, a few highly-utilized links can enormously increase the total congestion probabilities. Contrarily, when the amount of each traffic demand is small, the number of hops for each traffic demand will significantly affect the total congestion probabilities. Therefore, in order to minimize the total end-to-end congestion probabilities for all traffic demands, we need to minimize the total network utilization while limiting the maximum link utilization.

Based on the analysis above, we reformulate the path planning problem as follows:

Problem 1′ *Capacity-Bounded Shortest Path Planning*

Objective:

$$\text{minimize} \qquad \sum_{(i,j)\in E} \sum_{n} y_{ij}^{n} \tag{2.7}$$

Constraints:

$$\sum_{j} x_{ij}^{n}(m^n) - \sum_{j} x_{ji}^{n}(m^n) = \begin{cases} 1 & \text{if } i = \text{source} \\ -1 & \text{if } i = \text{destination} \\ 0 & \text{otherwise} \end{cases}, \ \forall i \in V,\ m^n \in M^n,\ n \in N$$

$$x_{ij}^{n}(m^n) = \begin{cases} 1 & \text{if } (i,j) \in P(m^n) \\ 0 & \text{otherwise} \end{cases}, \quad \forall (i,j) \in E,\ m^n \in M^n,\ n \in N$$

$$y_{ij}^{n} = \sum_{m^n} \bar{\rho}^{n}(m^n) x_{ij}^{n}(m^n), \quad \forall (i,j) \in E,\ n \in N$$

$$\sum_{n} y_{ij}^{n} \leq (1-\theta)C_{ij}, \quad \forall (i,j) \in E \tag{2.8}$$

Remarks:

- Objective function (2.7) aims to minimize the total network utilization.
- Constraint (2.8) tightens the upper bound on the link capacity of substrate network. θ is the capacity-limiting parameter.

Problem $1'$ is a capacity-bounded shortest path planning problem, which finds the optimal path set that minimizes the total network utilization under tightened capacity constraints. The problem has a form of minimum cost multi-commodity flow problem. This problem is known to be $\mathcal{NP}$-complete for integer flows [32], but there exist several approximation algorithms for solving the problem within polynomial time [33]. In this Chapter, we employ *branch-and-cut-and-price algorithm* [34], which is a combinatorial optimization method based on linear programming and integer relaxation.

Based on the problem above, we propose a heuristic to solve the substrate partitioning problem as follows:

The heuristic begins by computing the shortest path planning and the optimal capacity partitioning under the original capacity constraints of substrate network [Step 1−4]. Then it repeats to solve these two subproblems consecutively by reducing the capacity boundary by $\Delta\theta$ [Step 5−8]. The iteration stops when there is no more possible path set for the reduced capacity constraint [Step 14]. Finally, among the local solutions obtained at every iteration, the heuristic chooses the best solution that minimizes the total congestion externality $\sum_{\text{n}} \sum_{m^n} Q(m^n)$ [Step 9−13].

Algorithm 1 Heuristic for Optimal Substrate Partitioning

```
$\theta \leftarrow 0$
**Solve** Problem $1'$
**Solve** Problem 2
$MinCE \leftarrow \sum_{n} \sum_{m^n} Q(m^n)$
**repeat**
    $\theta \leftarrow \theta + \Delta\theta$
    **Solve** Problem $1'$
    **Solve** Problem 2
    **if** $\sum_{n} \sum_{m^n} Q(m^n) < MinCE$ **then**
        $OptPath \leftarrow$ solution of Problem $1'$
        $OptCap \leftarrow$ solution of Problem 2
        $MinCE \leftarrow \sum_{n} \sum_{m^n} Q(m^n)$
    **end if**
**until** solution of Problem $1' = \emptyset$
```

The heuristic basically takes into account the trade-off between the shortest path routing and load balancing. Shortest path routing minimizes the network utilization, but can cause some highly-utilized links, which make the link congestion probabilities extremely high. Tightening the capacity boundary through iteration benefits to release the traffic loads of these highly-utilized links. However, since the overloaded traffic demands can only be rerouted along longer (or equal-length) paths, the network utilization grows, and consequently the total congestion probabilities will increase. The heuristic provides an optimal point in this trade-off.

Note here that the heuristic is based on the proposition that the shortest path set is most likely to minimize the total congestion externality among the path sets that have the same maximum link utilization. On the assumption that this proposition holds, we can considerably reduce the search space of path planning problem. Strictly speaking, however, this proposition is not always true. Minimizing the network utilization which maintaining the same maximum link utilization leads to the minimum congestion probability, but it does not guarantee the minimization of total congestion externality, since each VN has different QoS requirement. Nevertheless, the heuristic will always provide a certain level of performance guarantee under any conditions, since it guarantees the low end-to-end congestion probability for every traffic demand evenly.

The computation time of the heuristic is directly proportional to the number of iterations, and the number of iterations is determined by the given traffic conditions and the size of $\Delta\theta$. If the value of $\Delta\theta$ is large, the iteration is repeated just a few times (e.g., if $\Delta\theta = 0.1$, the number of iterations is less than 10). A large $\Delta\theta$, however, decreases the accuracy of the heuristic due to the reduction of the search space. On the contrary, a small value of $\Delta\theta$ increases the computation time, but also increases the accuracy. Therefore, we need to choose an appropriate value of $\Delta\theta$ to satisfy the time and accuracy requirements.

2.6 Evaluation

In this section, we evaluate the performance of the proposed heuristic. We conduct two experiments: First, we examine the accuracy of the heuristic by comparison with the global optimum in a simple network topology. Then, we verify the efficacy and scalability of the heuristic for large network topologies and various traffic conditions.

Throughout the experiments, we employ COIN/BCP solver [35] in order to solve Problem $1'$, which implements the *branch-and-cut-and-price algorithm*

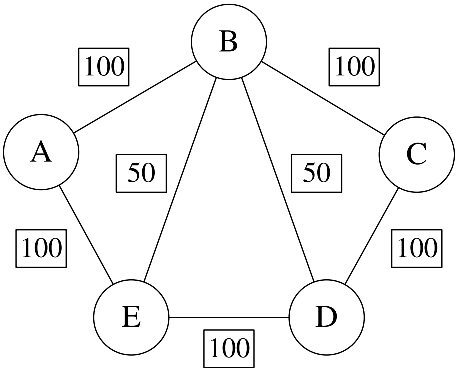


Figure 2.4: Substrate network topology for small network experiment.

for solving mixed integer programs. Then, we use *Newton's method* to solve Problem 2.

2.6.1 Small Network Experiment

First, we discuss the accuracy of the heuristic. Unfortunately, as mentioned earlier, since the substrate partitioning problem is $\mathcal{NP}$-complete, the computation time for the global optimum grows exponentially with the size of the substrate network and the number of traffic demands. Thus, separately from the other experiments, we consider a small-sized substrate network for the accuracy analysis of the heuristic, which allows us to obtain the global optimum in a reasonable time. First, we use a naïve search method to find out all possible path sets, and then solve the convex problems for all of them to obtain the optimal capacity partitioning. Finally, we reach the global optimum which minimizes the total congestion externality. For 10 end-to-end traffic demands, the total process takes less than 10 minutes.

We use a simple substrate network topology shown in Figure 2.4, which consists of 5 nodes and 7 links. The number in rectangle indicates the available

Table 2.1: Accuracy Rate of Heuristic (%)

$\Delta\theta$		scenario 1			scenario 2		
		0.03	0.05	0.15	0.03	0.05	0.15
mean	6	98.1	98.2	98.2	99.7	99.7	99.7
	8	96.2	96.2	64.9	100	100	100
	10	97.8	88.7	84.9	100	100	100
	12	100	87.6	73.5	100	100	100
	14	99.9	99.9	99.9	100	100	100
	16	99.9	99.9	72.0	100	100	100
	18	92.6	84.5	84.5	100	100	100
	20	100	77.1	56.1	100	100	100
avg		98.1	91.5	79.3	100	100	100

bandwidth capacity for each link. For simplicity, we assume that the substrate network is undirected (i.e., $\rho(A,B)=\rho(B,A)$), and partitioned into just two VNs. We randomly choose 5 source-destination node pairs for each VN, and assign them different amounts of average traffic demands, which initially range from 2 to 10. Then, we repeat the experiments by increasing the average traffic demands by 2. To investigate the effect of $\Delta\theta$ on the accuracy of the heuristic, we run the heuristic for different values of $\Delta\theta$ (i.e., 0.03, 0.05 and 0.15). The link utilization threshold η and the dissatisfaction index φ are set to 0.8 and 2, respectively.

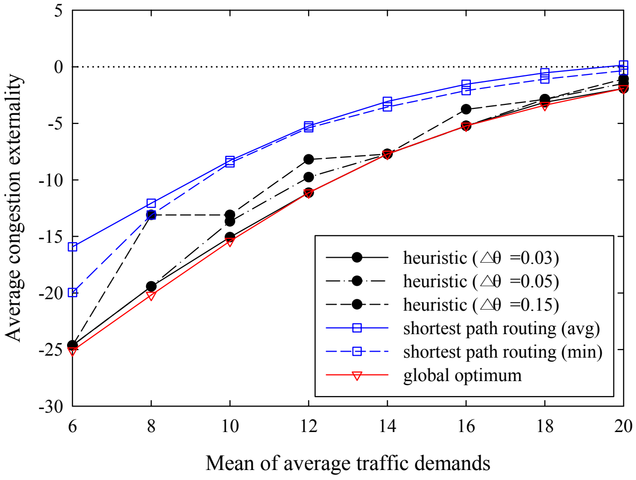
We perform the experiments for dozens of different scenarios. In order to assess the accuracy of the heuristic, we define the accuracy rate as the logarithmic ratio between the average congestion externality[6] of the heuristic and the global optimum. As a result, the average accuracy rates of all the scenarios

---

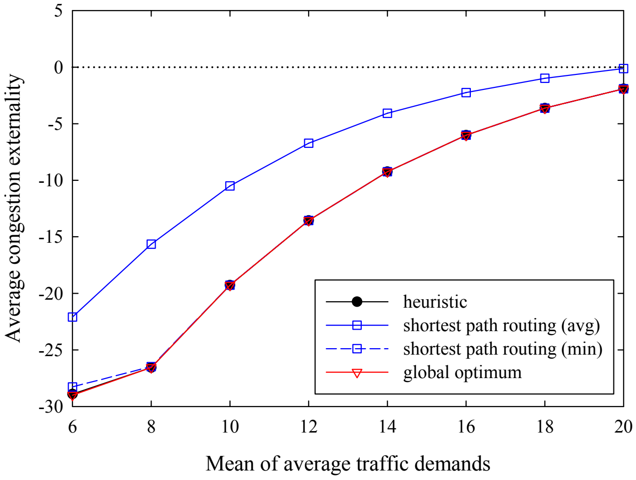
[6] The average congestion externality is given by $\bar{Q} = \frac{1}{m}\sum\limits_{m} Q$, where m is the total number of commodities without distinction of VN. The average congestion externality represents the degree of overall QoS guarantee.

we consider are calculated as about 99.2%, 94.8%, and 88.0% for $\Delta\theta = 0.03$, 0.05, and 0.15, respectively. Interestingly, the results of all the scenarios are grouped into two cases according to the pattern of accuracy rates for different values of $\Delta\theta$: In one case, the accuracy rate averagely decreases with an increase in $\Delta\theta$, but in the other case, the accuracy rate does not change with $\Delta\theta$. Table 2.1 shows the results from two sample scenarios, each of which represents one of the two cases. The difference between these two scenarios can be interpreted in a similar manner to the example illustrated in Figure 2.3. That is, in scenario 1, since the shortest path routing results in a few highly-utilized links, load balancing caused by the capacity boundary tightening reduces the total congestion externality, and more gradual tightening is likely to achieve better performance. On the other hand, in scenario 2, the shortest path routing leads to a balanced distribution of traffic demands; thus the optimal solution of the heuristic is generated in the first iteration (i.e., $\theta = 0$), and the value of $\Delta\theta$ does not affect the results at all.

Figure 2.5 shows the comparison of the results from the heuristic and the shortest path routing with the global optimum for scenarios 1 and 2. In the figures, for ease of comparison, we take the common logarithm of the average congestion externality. In case of the shortest path routing, since there exist more than one solution that satisfies the objective function, we present the mean value of the average congestion externality for all possible solutions as well as the minimum value. The x-axis represents the mean value of the average traffic demands for 10 commodities. As a result, Figures 2.5(a) and 2.5(b) show that the heuristic produces better or at least the same performance compared to the best result of the shortest path routing, and provides a good approximation to the global optimum for small value of $\Delta\theta$.



(a) scenario 1

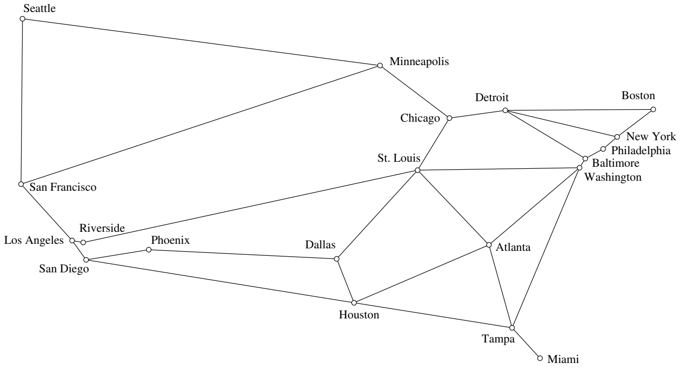


(b) scenario 2

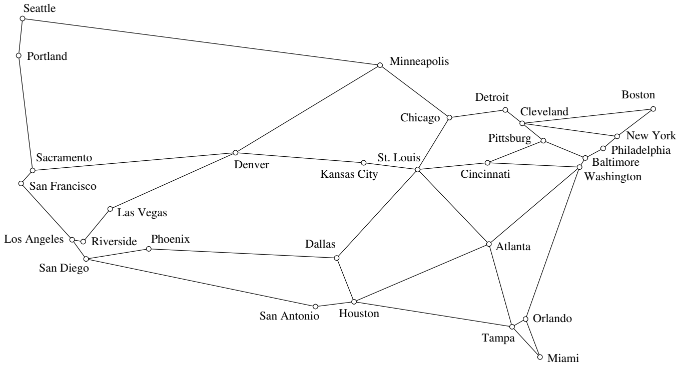
Figure 2.5: Comparison results of the proposed heuristic with other routing methods.

2.6.2 Large Network Experiment

In this section, we conduct extensive experiments under more various traffic conditions in order to verify the efficacy and scalability of the proposed heuristic. We consider two substrate network topologies, which are created by



(a) US_top20_MSA



(b) US_top30_MSA

Figure 2.6: Substrate network topologies for large network experiments.

selecting 20 and 30 largest metropolitan statistical areas (MSAs) from the US MSA rank by population in 2010, and randomly generating links between adjacent nodes. We assume that each link is bi-directional (i.e., directed graph), and every link capacity is equally set to 300. Figure 2.6 shows the substrate network topologies described above.

We assume that each substrate network is partitioned into two VNs, which have the congestion probability requirements of 0.1 and 0.2. We randomly choose 60 and 80 source and destination node pairs (i.e., 30 and 40 for each

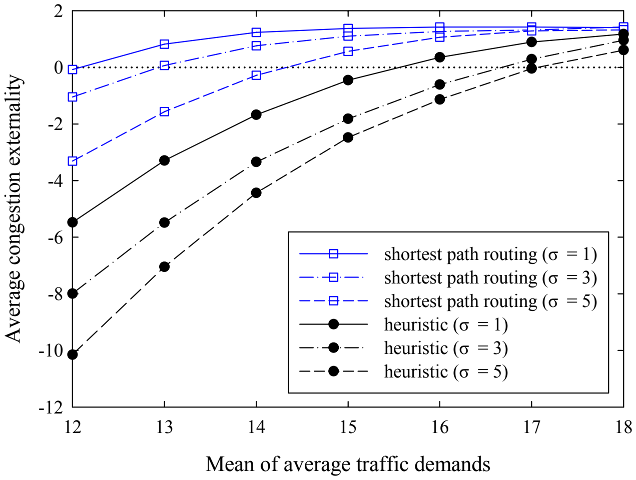
VN) in two topologies, respectively, and assign the same amount of traffic demands to both directions of each node pair. We assume that the average traffic demands follow a *Gaussian* distribution with mean m and standard deviation σ. We perform the experiments by increasing m for three different values of σ (i.e., $\sigma = 1, 3, 5$). The link utilization threshold η is set to 0.8, and the dissatisfaction index φ is set to 2.

Figures 2.7(a) and 2.7(b) show the comparison results of the proposed heuristic with the shortest path routing for two substrate network topologies in terms of the average congestion externality.[7] For ease of presentation, we use logarithmic scale for y-axis in the figures. The dotted line represents the normalized congestion probability requirement. Both figures show that the proposed heuristic outperforms the shortest path routing in terms of the average QoS guarantee. It seems that the larger the network scale becomes, the more noticeable the difference in QoS performance is. We can also see that the values of the average congestion externality for the heuristic show a strong tendency to become lower as σ grows. This tendency is due to the fact that diversifying the size of traffic demands enables more fine-grained load balancing. Additionally, from the results above, we can infer the maximum level of network utilization allowed to satisfy the QoS requirement. In the case of US_top30_MSA topology, for example, in order not to violate the given congestion probability requirements of VNs, we need to keep the network utilization less than about 40%.[8] This can provide the VN service providers

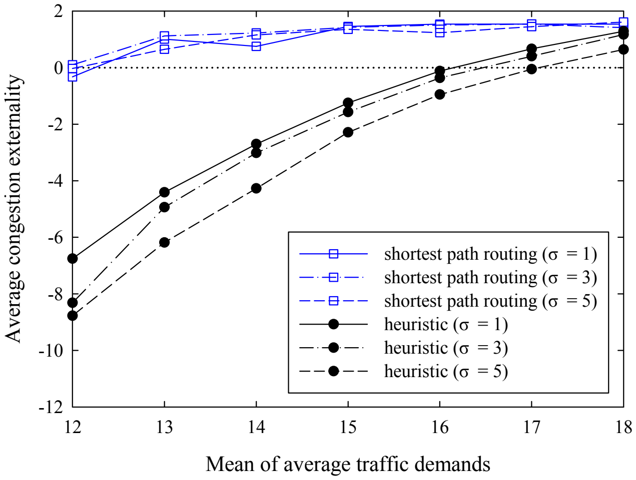
---

[7] For the given network and traffic conditions, since the shortest path routing algorithm can yield an infinite number of solutions, it is impossible to compute the mean and minimum value of the average congestion externality as in the 5-node experiments. Thus, we use the shortest path which is (randomly) chosen by COIN/BCP solver.

[8] We obtained almost the same value of maximum allowable network utilization in all values of σ.



(a) US_top20_MSA topology



(b) US_top30_MSA topology

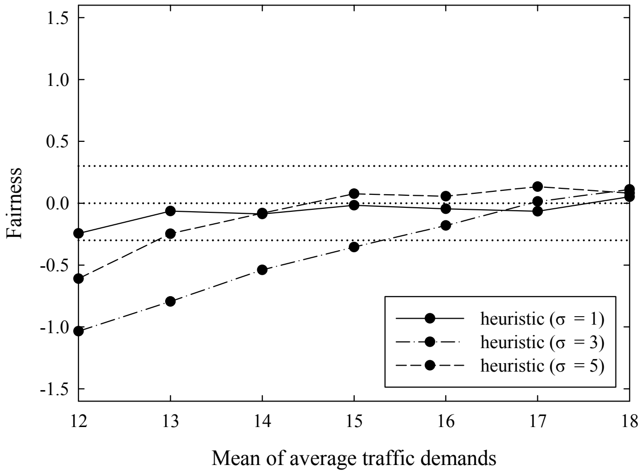
Figure 2.7: The average congestion externality depending on the change of mean and standard deviation of the average traffic demands.

with a good guideline for network planning or admission control.

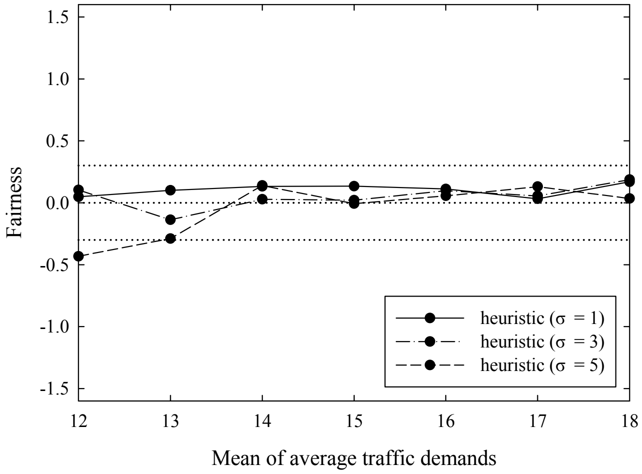
Figures 2.8(a) and 2.8(b) describe the fairness of the proposed heuristic for two topologies. In order to quantify the degree of differentiated QoS guarantee for two VNs which have different QoS requirements, we define a fairness index as the logarithmic ratio between the average satisfaction degrees of QoS requirement for two different VNs. In each figure, the dotted lines represent ×2, ×1, and ×0.5 in the ratio of satisfaction degree, respectively. The figures indicate that overall the values of fairness index do not change significantly according to m and σ, and most values are plotted between the ×2 and ×0.5 lines. Particularly in the US_top30_MSA topology, almost all values are concentrated near the ×1 line. Notice here that, even if the total traffic loads exceed the maximum allowable network utilization for QoS guarantee, the heuristic ensures fairness among VNs. From these results, we can verify that the heuristic achieves satisfactory performance even in more realistic network topologies and traffic conditions.

2.7 Summary

Network virtualization has mainly been deployed to build a testbed network because of its flexibility and independency of each VN. In this Chapter, we propose a new application of network virtualization to the layered network architecture offering differentiated services, and present a theoretical framework to design the NVE efficiently. Based on a stochastic traffic demand model and quantifiable QoS metric, we formulate the substrate partitioning problem using mixed integer programming, which provides an optimal resource allocation method including path planning and capacity partitioning. In order to solve the problem, we develop a simple heuristic based on the shortest path routing algorithm. The proposed heuristic outperforms other routing methods, and provides a reasonable approximation to the global optimum. In large networks experiments, the heuristic shows more fine-grained QoS differentiation.



(a) US_top20_MSA topology



(b) US_top30_MSA topology

Figure 2.8: Fairness depending on the change of mean and standard deviation of the average traffic demands.

However, there are several issues that remain unresolved in this study and require further study. First, since the substrate partitioning problem deals with the initial and planned resource allocation under the assumption that the average traffic demands between end nodes are stable for a long period, and the short-term fluctuation of traffic demands can be modeled as a stochastic process, it cannot provide an immediate and efficient solution to the unexpected events such as link failure, router crash and intentional network attack. Thus, an additional recovery method for coping with such unexpected events is required. We are currently working on developing a distributed rerouting and switching method for VNs, which cooperates with the substrate partitioning method. Analyzing the application of other QoS metrics and their mixed usage is another research topic that needs further attention. In this study, we only focus on the analysis of the substrate partitioning according to the congestion probability. Extending the proposed framework to support diverse QoS metrics will provide the service provider with a more efficient and profitable business model. Finally, the proposed heuristic admits of further improvements. To make the problem tractable, the heuristic does not take into account QoS differentiation in path planning step, which may not guarantee a satisfactory approximation to the global optimum in the worst-case situation. We leave the issue of developing a more sophisticated heuristic for future work.

Chapter 3

Optimal Pricing and Capacity Partitioning for Tiered Access Service in Virtual Networks

3.1 Introduction

With the diversification of applications, there has been a growing demand for tiered service in the Internet access market. A tiered service[1] allows the users to subscribe the best suited one to their needs among a set of tiers with gradually increasing prices. Since the tiered service provides various options of access services according to service level and access price, users are encouraged to spend their money more wisely, taking into account their needs and budgets. Furthermore, the tiered service enables Internet service provider (ISP) to improve the efficiency of network resources, which will eventually lead to the expansion of its market share in the fiercely competitive market. Because of these benefits to both user and ISP, many ISPs offer some forms of tiered Internet access service along with the corresponding multi-tiered price

---

[1] In this Chapter, unless otherwise noted, the term *service* refers to an access service, not an Internet application.

structure in current practice (e.g., \$ 10 per month for maximum download speed of 100 Mbps or \$ 3 for 10 Mbps).

Over the last decade, a large number of studies have been conducted on analyzing the economics of the tiered service [36–41]. However, a complex interaction between pricing, resource allocation, and user utility causes difficulty in developing a practical economic model for the tiered service. *Paris Metro pricing* (PMP) overcomes this difficulty in the way of using a multi-class flat-rate pricing scheme as a means of differentiating the service level [42]. Despite its simple structure, however, the PMP does not guarantee the average (or maximum) quality of service (QoS) for each of the service tiers, but rather determines the relative superiority among them [43]. Since the PMP cannot commoditize the service level of each tier, not only did it fail to attract users, but it has not been widely adopted in communication and service networks. Fundamentally, this limitation of the PMP scheme stems from the absence of a standard protocol for supporting the QoS guarantee in the current Internet. To address this problem, it will be necessary to reconsider the underlying protocol or network architecture for supporting the tiered service.

One possible approach to this problem is to utilize virtual networks (VNs). Network virtualization is an emerging technology for diversifying the Internet [3–5]. Since a network virtualization environment (NVE) accommodates different types of services within logically-partitioned VNs simultaneously and guarantees independence among them, the environment itself can serve as a layered network architecture for supporting tiered service (i.e., each VN provides different service level). Moreover, in the NVE, the physical resources such as router CPU and link bandwidth can be flexibly partitioned among

VNs, which facilitates a certain degree of QoS differentiation among service tiers. The VNs, therefore, can be an attractive solution to overcome the inherent limitations of single network in providing the tiered service.

In this Chapter, we aim to develop an economic model for a tiered service based on VNs.[2] First, in order to model the prevalent service tiering method in which each tier is priced differently according to the access rate (or maximum data transmission rate), we assume that different levels of *access rate regulation* are imposed over multiple service tiers. We analyze the impacts of this access rate regulation on user behavior and on the ISP revenue in both non-tiered and tiered services, and then derive a sufficient condition for stability of the user subscription dynamics. Finally, from this analysis, we determine the optimal pricing and capacity partitioning in terms of the ISP revenue for the tiered service in the VN architecture.

This Chapter presents three main contributions. First, we present a new network economics model to account for the user subscription dynamics with the access rate regulation in a tiered service based on VNs. Specifically, we derive a sufficient condition for the stability of user subscription dynamics in non-tiered and tiered services. Second, we formulate the revenue maximization problem of the tiered service in the VNs as an optimization problem which simultaneously determines the optimal pricing and capacity partitioning. In this problem, the capacity for each service tier as well as the access price are designated as a decision variable, thus leading to more flexible and efficient operation of the tiered service. Finally, we consider a more realistic user utility model for the tiered service by applying the access rate regulation to

---

[2] This work was performed in collaboration with Prof. Han-Yoo Jeong and Prof. Seung-Woo Seo.

the analysis. To the best of our knowledge, this is the first study to present a theoretical analysis for the impacts of the access rate regulation on the user behavior and on the ISP revenue in the tiered service. Our economic model and analysis in this study will provide a useful benchmark for deploying a tiered Internet access service.

The rest of this Chapter is organized as follows: Section 3.2 presents a motivating example that illustrates the impacts of access rate regulation on the user behavior and on the ISP revenue. In section 3.3, we describe the tiered service model and the assumptions used in this study. Sections 3.4 and 3.5 present the analysis on the optimal pricing and capacity partitioning for non-tiered and tiered services, respectively. Section 3.6 provides numerical results that demonstrate the results of our analysis. Then, we highlight the contributions of this section by comparing them with those of the related work in section 3.7. Finally, we summarize in section 3.8 with further discussion.

3.2 Motivating Example

In this section, with a simple conceptual example, we first illustrate how the access rate regulation affects the user behavior and ISP revenue, and then briefly compare the maximum ISP revenues of non-tiered and tiered services under the assumption of the access rate regulation.[3]

Let us consider a situation where an ISP offers Internet access service to its subscribers through a 10 Gbps optical fiber link. Since multiple users share the same bandwidth, the access rate for an individual user depends upon the total

---

[3] Configurations and parameters in this section are the same as those in the section 3.6 unless otherwise specifically stated.

number of subscribers. In Figure 3.1(a), the dashed line represents the access rate for each user according to the number of subscribers. Initially, each user makes a decision as to whether to join the service or not based on the given access price. Then, the total number of subscribers determines the access rate for each user. Since different users may have different satisfaction for the same access rate, some of them decide to opt out of the service, while the others retain it. The decrease in the number of subscribers leads to the improvement of the access rate for each user, which again promotes more users to join the service. This fluctuation in the number of subscribers continues over time until it eventually converges to a fixed point, called a *steady-state point.* At this point, we consider the ISP revenue from its service which is defined as the product of its access price and the steady-state number of its subscribers. Notably, despite the trade-off between the access price and the steady-state number of subscribers, the ISP revenue can be seen as a nondecreasing function of the access price due to the inelastic subscription of the users, which will be shown in the later.

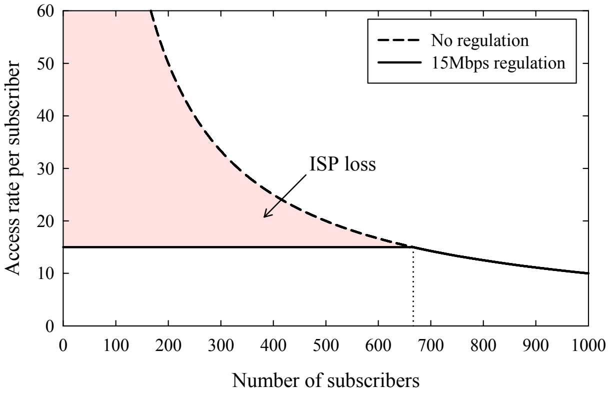
In practice, however, a user cannot exploit the fair share of the access rate when the number of subscribers is small. This is because the access rate is limited by several external factors such as the line speed of access link and the policy of ISP.[4] Thus, it is more realistic to assume that the access rate follows a truncated curve as depicted by solid line in Figure 3.1(a). The *effective* access rate of each user is initially a constant up to a certain number of subscribers, and then a decreasing function along the original curve. In fact, this regulation has an adverse effect on the user subscription and the ISP revenue; compared

---

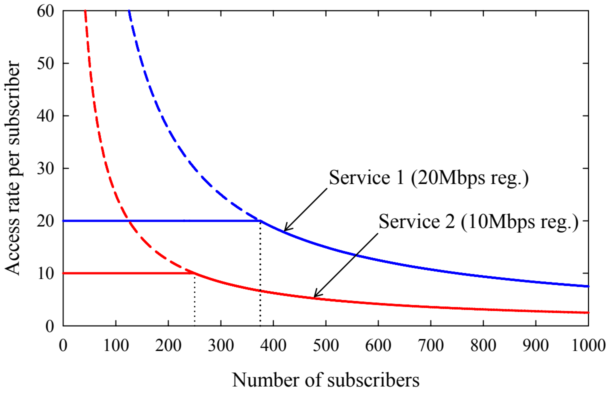
[4] We assume that ISPs can regulate the access rate of users below a certain level in order to support their tiered service policy.

with the same service *without* regulation, fewer users will join the service due to its low access rate, which results in a reduction of the ISP revenue. It is intuitively clear that the higher the access price, the smaller the steady-state number of subscribers, and therefore the greater the difference between the ideal and actual steady-state numbers of subscribers. Indeed, the ISP revenue against the access price can be seen as a bell-shaped curve (as will be shown in Figure 3.5(c)), which is comparable to the result of the ideal case. Thus, we can obtain the maximum ISP revenue and the access price at that time. For example, the ISP revenue for the regulation rate of 15 Mbps has a maximum value of \$ 25,669 at the access price of \$ 38.6, which will be addressed in section 3.6.

Let us now consider a tiered service example, in which an ISP provides two-tier Internet access service over dually-partitioned VNs: one for high speed access at a premium price, and the other for relatively lower speed access at a reasonable price. We assume that the 10 Gbps physical link is partitioned into 7.5 Gbps and 2.5 Gbps virtual links, which are completely isolated from each other, and that the access rates for two services are regulated to 20 Mbps and 10 Mbps, respectively, in order to differentiate the service level. Figure 3.1(b) shows the effective access rates for two services. In this simple two-tier service example, each user selects one service to join by comparing the current access rates of both services or decides not to subscribe any service at all. As in the case of the non-tiered service, we can find the maximum value of the ISP revenue and the set of access prices for two services at that time. From the results in section 3.6, the maximum ISP revenue is equal to \$ 26,194.4 when the access prices are \$ 64.8 and \$ 31.6.



(a)



(b)

Figure 3.1: Access rate regulation for (a) non-tiered service and (b) tiered service.

To summarize, the tiered service can achieve a slightly higher ISP revenue than the non-tiered service. Even though different regulation rates are imposed on two examples, of particular interest is the observation that, with the access rate regulation, a service differentiation can lead to more benefits than a single service, which is contrary to the common belief in this field. This observation motivates us to analyze the effects of the access rate regulation on the user behavior and the ISP revenue in both non-tiered and tiered services in more detail.

3.3 A Tiered Service Model

In this section, we describe a tiered service model and the assumptions used in our analysis. In the tiered service, users can select the best service suited to their needs from a small set of service tiers at gradually increasing pricing [36–41]. In communication networks, the tiered service is usually implemented by the Diffserv and the MPLS architecture. To enhance the QoS differentiation and the survivability of the tiered services, in this Chapter, we focus on the implementation of the tiered service based on the *network virtualization* [3–5].

3.3.1 Network Virtualization Environment

We first describe the NVE to support the tiered access service. As shown in Figure 3.2(a), users have access to the Internet through an ISP network. The ISP network is composed of a set of (virtual) routers interconnected with each other, aggregating different levels of user traffic, and routing them towards the Internet backbone. We assume that this ISP network is virtualized into m logical networks (i.e., VNs),[5] each of which offers a different service level.[6] Thus, as in Figure 3.2(b), traffic from users who are subscribing to service tier i can be forwarded only over i'th VN. Since each VN is completely isolated from the others, packets traveling over a VN cannot be switched to another VN, and a congestion occurring within a VN does not affect the performance of the others. Thanks to the network virtualization, the capacity of each VN

---

[5] We only consider the virtualization of the *shared* network, not of the access link dedicated to each user.

[6] Our vision on network virtualization that serving different service levels over different VNs is similar to that in [44], albeit different in its application.

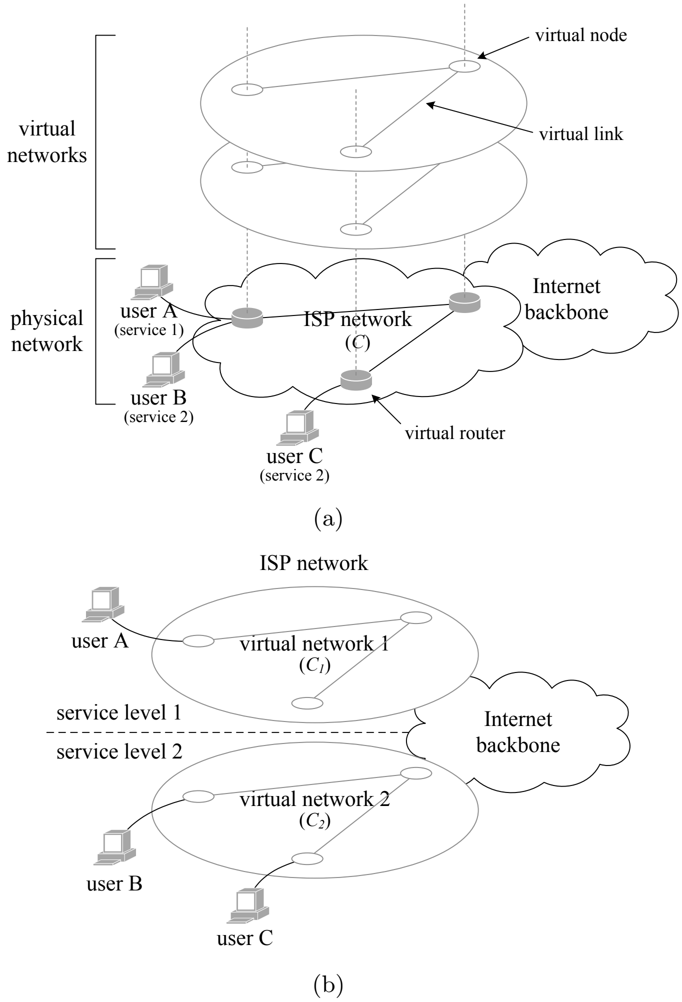


Figure 3.2: ISP network for tiered access service. (a) shows the scope of network virtualization, and (b) shows how the virtual networks support the tiered access service.

can be shared only by users subscribing to the same service tier. Thus, even in congested condition, users are assured of the same access quality within the same service tier, while experiencing differentiated access quality from those of other VNs (i.e., other service tiers).

In this study, we consider a general form of VN, which is a group of virtual nodes connected together via a set of virtual links. In a VN, each virtual node is a logical partition of the physical resources (e.g., CPU and memory), hosted on a virtual router; each virtual link is (in general) a logical tunnel (e.g., IP-in-IP, IPsec, and MPLS) over one or multiple physical links between two virtual nodes, occupying a certain portion of bandwidth resources. We assume that each service tier has its own ID, written on every packet; so at each gateway (virtual) router, incoming packets from users are classified and forwarded into the corresponding VNs according to this ID. Further details on the technical issues of network virtualization are beyond the scope of this study.[7]

We now model the capacity partitioning of the ISP network. For ease of analysis, we view the whole ISP network as a single link with bandwidth capacity C, which can be considered as the total upstream capacity boundary of the network (i.e., the sum of simultaneous upstream capacity of all links, or simply the capacity of an optical fiber cable connected to the Internet backbone). Let $\theta_i \in (0, 1)$ denote a fixed partitioning coefficient. Then, the capacity of i'th VN is equal to $C_i = \theta_i C$, where $\sum_{i=1}^{m} \theta_i = 1$. This simplified network model, in fact, cannot fully characterize the contribution of each of the components and their networking to the network performance. In this study, however, of importance is only the thickness of the end-to-end pipeline (i.e., end-to-end bandwidth) assigned to each user, not its length or path. Thus, in this context, it is reasonable to model the network as the bunch of pipelines which has a certain capacity boundary. This single capacity assumption has

---

[7] Our assumptions on network virtualization in this Chapter are generally applicable to all types of VN testbeds described in [14].

been widely adopted in previous studies in the area of network economics in order to focus on capacity sharing among users [40–42, 45–50].

Meanwhile, as briefly stated in section 3.2, we consider the *last mile* regulation of access quality. In other word, even if the number of users is small so that enough capacity is available in the ISP network, the access rate of each user is bounded at the access link by the maximum data rate of the access technology, such as dial-up, ISDN, cable modem, and xDSL. Here we further assume that, in order to build a tiered access service structure, the ISP can restrict the access rate to a lower value than the data rate of the access link by configuring either an access modem (e.g. cable modem) or an access router (e.g. xDSL). In the next section, we model the effective access rate considering this *last mile* regulation.

3.3.2 Effective Access Rate

In general, the Internet access quality can be characterized by diverse network performance measures such as throughput, delay, jitter, loss rate, and so on. However, most users have difficulty in assessing the access quality due to the lack of information about how to measure and analyze them. In this regard, the most simple and practical measure is *effective access rate* of each user. Thus, in this study, we assume that each user decides which service to subscribe to, based on the effective access rate and the price for each access service.

The effective access rate is closely related to the number of users. Let N denote the number of potential users who can possibly subscribe to an Internet access service. The number of users who subscribe to service i $(0 \leq i \leq m)$

is given by n_i, where n_0 stands for the number of *unsubscribed* users, i.e., $\sum_{i=0}^{m} n_i = N$. Then, the effective access rate q_i is given by

$$q_i = \frac{C_i}{n_i}. \tag{3.1}$$

Note here that the effective access rate depends on the link capacity provisioned to each service. Thus, capacity partitioning over multiple service tiers has a dominant effect on differentiating the service level. We will study the optimal capacity partitioning for the maximization of the ISP revenue in the later sections.

Recall that the effective access rate is regulated by several external factors. Let r_i denote the regulated access rate of service i. Then, the effective access rate for service i can be rewritten by

$$q_i = \begin{cases} r_i, & \text{if } n_i \le C_i/r_i \\ C_i/n_i, & \text{otherwise.} \end{cases} \tag{3.2}$$

Since the regulation rate is a variable determined by the data rate of access technology or the operational policy of each ISP, it is probably not a controllable parameter. Thus, in this Chapter, we restrict our focus on the effects of other variables including capacity and price under the given regulation condition.

3.3.3 Valuation Parameter and User Utility

In this study, we consider a heterogeneous user model in the sense that each user may have a different *valuation* for the same service quality (i.e., effective access rate). Each user j is characterized by a non-negative valuation parameter v_j which represents the maximum willingness to pay for the *unit* effective access rate of a service. The valuation parameter v_j can be modeled by a *value* of a random variable V defined in a positive real number $V \in \mathbb{R}_+$. We assume that the valuation parameter of each user does not change over time. That is, user j assesses the (monetary) value of service i as $v_j q_i$ (> 0).[8] Then, we can denote the cumulative distribution function (CDF) of the valuation parameter by $F_V(x) = \Pr(V \leq x)$. The corresponding probability density/mass function can be denoted by $f_V(x)$. Most of the related works in the literature have considered a simple uniform distribution for the valuation parameters [45, 46, 51]. Contrary to these works, in the study, we consider a general bell-shape curve for the CDF of random variable V. For example, we consider the *truncated* normal distribution as a particular example of the CDF of the valuation parameter, i.e.

$$F_V(x) = \begin{cases} 0, & \text{if } x \leq 0 \\ \int_{-\infty}^{x} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\frac{(v-\mu)^2}{\sigma^2}} dv, & \text{otherwise.} \end{cases} \tag{3.3}$$

In order to limit the impacts of the truncation, we impose an additional constraint that the parameter μ must be much greater than σ, i.e., $\mu \geq 3\sigma$.

We can now define each user's utility for a service. When user j joins service i, its utility is given by

$$u_{ij} = v_j q_i - p_i \tag{3.4}$$

where p_i is the access price for service i.

In this study, we consider a flat-rate pricing, where the price p_i is initially determined by the ISP and does not vary with time or the service quality.

---

[8] Note here that the valuation parameter v_j is not indexed by i. Each user's valuation criteria for different types of service are clearly different (e.g., VDSL and 3G). However, in this Chapter, the index i does not indicate the type of service, but rather refers to the service level. Thus, it is reasonable to assume that v_j is the same for different values of i.

Notice that, in reality, a user of a service has a barrier against the subscription to another service, due to the cost of customer-provided equipment (CPE), the stipulated time of the service, the ISP brand loyalty, etc. However, it is very difficult to account for these factors in the network economics model. To simplify this problem, we make the two additional assumptions:

- *Zero subscription and secession barrier*: Each user has a freedom to choose the service to subscribe.
- *Perfect information*: Each user has the perfect information about access rate q_i of all services at the previous time.

Under the above assumptions, user j makes a rational decision as to which service to subscribe, or whether to retain the service. When there are options of services available, the user will decide to join the service that gives the *highest* utility value. In other words, if the user j subscribes to the service k, then

$$k = \underset{i \in \{0, \cdots, m\}}{\arg\max} \; u_{ij}, \tag{3.5}$$

where service 0 represents the case in which user j does not subscribe to any services, i.e., $u_{0j} = 0$ for all j. In this Chapter, we denote by g_k, the probability that a user selects the service k. If none of the services have positive utility (i.e., $u_{ij} \leq u_{0j} = 0$ for all $i \in \{1, \cdots, m\}$), the user will not opt into any services at all. This type of utility function based on each user's rational behavior is widely used in the literature of economics [46–48, 52]. In some literature, the negative form of utility function is employed to highlight the effect of congestion externality [45, 49–51]; however, it can be easily transformed to the positive form as (3.4).

3.3.4 User Subscription and the ISP Revenue

The collection of subscription decisions of all users determines the total number of subscribers for each service. To take into account the user subscription dynamics[9], we consider a *discrete-time* utility model in which N potential users make a rational decision to subscribe a service at each time point t based on the effective access rate of each service at time $t-1$. We can represent the user subscription dynamics at time t by a $(m+1)$-tuple vector $\mathbf{n}(t) = [n_0(t), n_1(t), \cdots, n_m(t)]$, where $n_i(t)$ is the number of subscribers of service i at time t, i.e. $\sum_{i=0}^{m} n_i(t) = N$. Depending on the price values, the user subscription dynamics may converge to a steady-state point $\mathbf{n}^* = [n_0^*, n_1^*, \cdots, n_m^*]$, where n_i^* is the steady-state number of subscribers of service i.

Denoting the $(m+1)$-tuple price vector by $\mathbf{p} = [0, p_1, p_2, \cdots, p_m]$, the total revenue of the ISP is defined as the inner-product of the steady-state subscription vector and the corresponding price vector as follows:

$$R = \mathbf{n}^* \cdot \mathbf{p} = \sum_{i=1}^{m} n_i^* p_i. \tag{3.6}$$

In general, the number of subscribers decreases with an increase in the access price. Taking into account this trade-off between p_i and n_i^*, we examine the optimal pricing for both non-tiered and tiered services in the following two sections.

---

[9] The term *user subscription dynamics* is first used in [46] to describe the user behavior for the subscription to a service. We adopt this term and partly its analytical framework to our study.

3.4 Non-tiered Service Analysis

In this section, we analyze the impacts of access rate regulation on the ISP revenue and on the pricing strategy in a non-tiered Internet access service.

3.4.1 User Subscription Dynamics

In this section, we study the dynamics of user subscription to a non-tiered service. We first derive a set of equations whose steady-state fixed point can be obtained by repeated substitutions. Next, we examine the condition that the set of equations does not converge to a fixed point in the steady state.

As we stated before, the utility of user j subscribing to the non-tiered service can be formulated in terms of the effective access rate at the previous time period as follows:

$$u_j(t) = v_j q(t-1) - p. \tag{3.7}$$

Here, we dropped the service index i for the simplicity of the expression.

A rational user will join the non-tiered service, if the subscription maximizes the user's utility function, i.e. $u_j(t) \geq 0$. In (3.7), this inequality implies that the joining user's valuation must be at least the *unit-rate price (URP)* $p^{UR}(t)$ of the non-tiered service, i.e., $v_j \geq p^{UR}(t) = p/q(t-1)$. Then, the *subscription probability* $g(t)$ that a user joins the non-tiered service at time period t can be expressed as a function of the CDF of the valuation parameter, i.e.,

$$g(t) = \Pr\left(V \geq p^{UR}(t)\right) = 1 - F_V\left(\frac{p}{q(t-1)}\right). \tag{3.8}$$

Using (3.8), the number of subscribers at time t is represented by

$$n(t) = Ng(t) = N\left[1 - F_V\left(\frac{p}{q(t-1)}\right)\right]. \tag{3.9}$$

Taking into account the regulation in (3.2), the effective access rate $q(t)$ of the non-tiered service is given by

$$q(t) = \begin{cases} r, & \text{if } n(t) \leq C/r \\ C/n(t), & \text{otherwise.} \end{cases} \tag{3.10}$$

As t increases, the user subscription dynamics in (3.9) and (3.10) can be calculated by repeated substitutions. However, if the number of subscribers converges to a steady-state point n^*, this point does not change over time, i.e.,

$$n^*(t+1) = n^*(t). \tag{3.11}$$

In the *regulated range* satisfying $n^* \leq C/r$, the steady-state point n^* is obtained by combining the set of equations in (3.9)-(3.11),

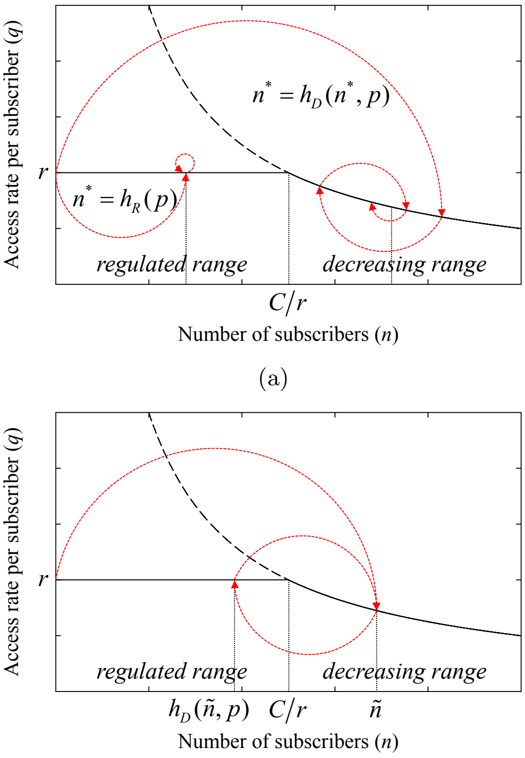
$$n^* = h_R(p) = N\left[1 - F_V\left(\frac{p}{r}\right)\right] \leq \frac{C}{r}, \quad \text{if } n^* \leq \frac{C}{r}. \tag{3.12}$$

Similarly, in the *decreasing range* satisfying $n^* > C/r$, we have

$$n^* = h_D(n^*, p) = N\left[1 - F_V\left(\frac{pn^*}{C}\right)\right] > \frac{C}{r}, \quad \text{if } n^* > \frac{C}{r}. \tag{3.13}$$

For a given access price, if either of the conditions in (3.12) and (3.13) is satisfied, the number of subscribers always converges to a steady-state point n^*. Otherwise, the user subscription dynamics does not converge to a fixed point, but oscillates indefinitely around it.

For the ease of understanding, we illustrate the examples of the user subscription dynamics in the converging and the oscillating conditions in Figures 3.3(a) and 3.3(b), respectively. In both figures, the initial number of subscribers is set to zero, i.e. $\mathbf{n}(0) = [N, 0]$. Figure 3.3(a) shows examples of two converging conditions (3.12) and (3.13). Since the procedures are straightforward, we omit a detailed description here. Figure 3.3(b) shows an example



(a)

(b)

Figure 3.3: Examples of user subscription dynamics in (a) convergence and (b) oscillation conditions.

of the user subscription dynamics when neither condition (3.12) nor (3.13) is satisfied. In this figure, $\tilde{n}$ indicates the number of subscribers obtained from the regulated access rate (i.e., $\tilde{n} = h_R(p)$). If $\tilde{n}$ belongs to the *decreasing range* (i.e., $\tilde{n} > C/r$), and the subsequent number of subscribers belongs to the *regulated range* again (i.e., $h_D(\tilde{n}, p) \leq C/r$), the user subscription dynamics oscillates repeatedly between the two values, one in the *regulated range* and the other in the *decreasing range*. This phenomenon fundamentally stems from the following two facts: 1) The access rate falls into two distinct ranges; and 2) the regulated access rate always leads to the same number of subscribers in the decreasing range at the next time period.

From this observation, we present a necessary and sufficient condition for the oscillation of user subscription dynamics in the following lemma:

Lemma 1. *For a non-tiered Internet access service whose access rate is regulated to r, the number of subscribers for the service does not converge to a steady-state value, but oscillates indefinitely, if and only if*

$$\tilde{n} > \frac{C}{r} \quad \text{and} \quad h_D(\tilde{n}, p) \leq \frac{C}{r}$$

where $\tilde{n} = h_R(p)$.

Manipulating the above equations, we can derive the condition of the access price for the oscillation as follows:

$$p_l = \frac{C}{\tilde{n}} F_V^{-1}\left(1 - \frac{C}{Nr}\right) \leq p < p_h = r F_V^{-1}\left(1 - \frac{C}{Nr}\right).$$

Notice that the upper ($p \geq p_h$) and the lower ($p < p_l$) ranges of the access price for the steady-state solutions are associated with the *regulated* and the *decreasing ranges* in Figure 3.3, respectively. Thus, if $p \geq p_h$ or $p < p_l$, the user subscription dynamics can be stabilized to the steady state in (3.12) or (3.13), respectively, after some iterations.

3.4.2 Optimal Pricing for Maximizing the ISP Revenue

We now discuss the optimal pricing that maximizes the ISP revenue. In (3.6), the ISP revenue for a non-tiered service is given by the product of the steady-state number of subscribers and the corresponding access price, i.e., $R = n^* p$.

Since the oscillation in the number of subscribers does not guarantee the stability of the ISP revenue, we consider only the condition under which the user subscription dynamics converges to a fixed point. Recall that the price condition for the convergence falls into either of the two distinct ranges, i.e. the *regulated* and the *decreasing ranges*. Between the two ranges, the ISP revenue of the lower price range belonging to the *decreasing range* satisfies the following property:

Lemma 2. *If the steady-state number of subscribers satisfies* $n^* > C/r$ *and* $n^* = N\left(1 - F_V\left(pn^*/C\right)\right)$*, the ISP revenue given by* $R = n^* p$ *is an increasing function of* p *regardless of* $F_V(\cdot)$*.*

Since the ISP revenue is an increasing function of the access price p, it has a maximum value when the access price has the upper bound value (i.e., $p = p_l$). From this property, we can further derive the following lemma on the ISP revenue and the optimal pricing:

Lemma 3. *The maximum ISP revenue for a non-tiered service always occurs in the range where the access rate is regulated.*

This result does not seem to capture the intuition since the access rate regulation gives rise to some loss of the ISP. Intuitively, the optimal steady-state number of subscribers that maximizes the ISP revenue would exist around the contact point between the *regulated* and *decreasing ranges* (i.e., $n = C/r$). However, the oscillation of user subscription dynamics restricts the condition for obtaining stable revenue around this point (see Figure 3.5(c)), thus leading to the above result. More detailed explanations are given by the proofs of Lemmas 2 and 3 in the Appendix.

Based on the above lemmas, we can restrict the search space for the optimal access price to the *regulated range*. Then, we can formulate the revenue maximization problem for a non-tiered service as follows:

Problem 1.

$$\begin{array}{ll} \textit{maximize} & Np\left(1-F_V\left(\frac{p}{r}\right)\right) \\ \textit{subject to} & \\ & p \geq rF_V^{-1}\left(1-\dfrac{C}{Nr}\right) \\ \textit{variable} & p \end{array}$$

We now apply the general analysis on the optimal pricing for the non-tiered service to the particular case where the valuation parameter follows a truncated normal distribution in (3.3).

Problem 1-1.

$$\begin{array}{ll} \textit{maximize} & \frac{Np}{2}\left(1-\mathit{erf}\left(\frac{p/r-\mu}{\sigma\sqrt{2}}\right)\right) \\ \textit{subject to} & \\ & p \geq r\left(\mu+\sigma\sqrt{2}\ \mathit{erf}^{-1}\left(1-\dfrac{2C}{Nr}\right)\right) \\ \textit{variable} & p \end{array}$$

Here, $\mathit{erf}\,(x)$ is the Gaussian error function defined as $\mathit{erf}\,(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt$.

For the error function and its inverse function, we have the Taylor series expansion:

$$\mathit{erf}\,(x) = \frac{2}{\sqrt{\pi}}\left(x-\frac{x^3}{3}+\frac{x^5}{10}-\frac{x^7}{42}+\cdots\right)$$

$$\mathit{erf}^{-1}(x) = \frac{\sqrt{\pi}}{2}\left(x+\frac{\pi}{12}x^3+\frac{7\pi^2}{480}x^5+\frac{127\pi^3}{40320}x^7+\cdots\right)$$

By using these, we can simplify the problem into a polynomial form, and obtain the approximations for some physically meaningful values. First, the boundary price p_h is directly derived from the constraint as follows:

$$p_h = r\left(\mu+\sigma\sqrt{\frac{\pi}{2}}\left(\left(1-\frac{2C}{Nr}\right)+\frac{\pi}{12}\left(1-\frac{2C}{Nr}\right)^3\right)\right) \tag{3.14}$$

Next, we can obtain the vertex price $\hat{p}$ from the first derivative of the objective function [53]. Since the vertex price $\hat{p}$ satisfies that

$$1 - F_V\left(\frac{\hat{p}}{r}\right) - \frac{\hat{p}}{r} f_V\left(\frac{\hat{p}}{r}\right) = 0, \tag{3.15}$$

we can compute the approximation of $\hat{p}$ by solving the low-order terms of the following equation:

$$\sum_{k=0}^{\infty} \frac{(-1)^k}{k! 2^k} \left(\frac{\hat{p}/r - \mu}{\sigma}\right)^{2k} \left(\hat{p}/r + \frac{\hat{p}/r - \mu}{2k+1}\right) = \sqrt{\frac{\pi}{2}}\sigma. \tag{3.16}$$

Then the maximum ISP revenue is generated at either of these prices. For a given condition, if the value of p_h is smaller than $\hat{p}$, the ISP revenue has the maximum value at $\hat{p}$, and vice versa. Thus, we can easily obtain the optimal price and its revenue by applying the given parameters to the equations (3.14) and (3.16), without the need to solve Problem 1-1.

3.5 Tiered Service Analysis

In this section, we extend our analysis to an ISP which provides two-tier Internet access service over dually-partitioned VNs: Service 1 has higher-speed access (i.e., higher regulation rate) with expensive price than service 2, i.e.,

$$r_1 > r_2 \text{ and } p_1 > p_2. \tag{3.17}$$

3.5.1 User Subscription Dynamics

We first model the user subscription dynamics for the tiered service based on the discrete-time utility model in section 3.3.4. As we stated before, the valuation parameter of user j can be seen as a random value in $\mathbb{R}_+ = (0, \infty)$.

Depending on its valuation parameter v_j, user j chooses the best service maximizing its utility at each time period t among three subscription options in (3.5): not to subscribe to any services ($k = 0$), or to subscribe to either service 1 ($k = 1$) or service 2 ($k = 2$). At each choice, the utility of user j joining service k is

$$u_{kj}(t) = \begin{cases} 0, & \text{if } k = 0 \\ v_j q_k(t-1) - p_k, & \text{otherwise.} \end{cases} \tag{3.18}$$

As we discussed in section 3.4.1, user j subscribes to service k if the valuation parameter v_j is at least the URP of the service, i.e., $v_j \geq p_k^{UR}(t) = p_k/q_k(t-1)$. In the two-tier service, we impose an additional requirement that the utility of the service must be greater than or equal to that of the other service, i.e., $u_{kj}(t) \geq u_{k'j}(t)$ where $k, k' \in \{1, 2\}$ and $k \neq k'$. Here, we define the *service selection threshold (SST)* V_T as the threshold value of valuation parameter v_j that makes the utility of both services equal ($u_{1j}(t) = u_{2j}(t)$)

$$V_T q_1(t-1) - p_1 = V_T q_2(t-1) - p_2 \Rightarrow V_T = \frac{p_1 - p_2}{q_1(t-1) - q_2(t-1)}. \tag{3.19}$$

Basically, if the value of valuation parameter v_j is larger than this SST value, i.e., $v_j > V_T$, user j will choose service 1; on the contrary, if $v_j < V_T$, the user will choose service 2. However, the detailed characteristics of the user subscription depend not only on the effective access rates of the two services at the previous time period ($q_1(t-1)$ and $q_2(t-1)$), but also on the URPs of the two services ($p_1^{UR}(t)$ and $p_2^{UR}(t)$). Thus, we need to take into account the different combinations of these conditions very carefully. Below, we will examine the details of the best subscription strategy of user j in all possible cases.

Case $\mathcal{A}$: $q_1(t-1) \geq q_2(t-1)$ and $p_1^{UR}(t) \geq p_2^{UR}(t)$

We first consider case $\mathcal{A}$ in which the effective access rate and the URP of service 1 are greater than or equal to those of service 2, respectively, i.e., $q_1(t-1) \geq q_2(t-1)$ and $p_1^{UR}(t) \geq p_2^{UR}(t)$. In this case, the SST V_T is greater than the URP of service 1, i.e.,

$$V_T \geq p_1^{UR}(t) \geq p_2^{UR}(t) \geq 0. \tag{3.20}$$

Figure 3.4(a) illustrates case $\mathcal{A}$ in which user j chooses *every* subscription option depending on its valuation parameter v_j. Since $q_1(t-1) \geq q_2(t-1)$ and $V_T \geq p_1^{UR}(t)$, user j selects service 1 if $v_j \geq V_T$. Similarly, if $p_2^{UR}(t) \leq v_j \leq V_T$, the utility of service 2 becomes the largest value. Finally, if $v_j \leq p_2^{UR}(t)$, the utilities of both services become negative, which leads user j not to subscribe to any services. To summarize, the subscription probabilities that a user joins service i at time t in case $\mathcal{A}$ are given by

$$g_0^{\mathcal{A}}(t) = F_V\left(p_2^{UR}(t)\right),\ g_1^{\mathcal{A}}(t) = 1 - F_V\left(V_T\right),\ \text{and}\ g_2^{\mathcal{A}}(t) = F_V\left(V_T\right) - F_V\left(p_2^{UR}(t)\right), \tag{3.21}$$

where $g_0^{\mathcal{A}}(t)$ refers to the probability that a user does not join any services at time t.

Example 1-$\mathcal{A}$ *Consider a two-tier Internet access service in which* $p_1 = 20$, $p_2 = 8$, $q_1(t-1) = 5$, *and* $q_2(t-1) = 4$. *Then, the inequality in (3.20) is satisfied as follows:* $V_T = 12 \geq p_1^{UR}(t) = 4 \geq p_2^{UR}(t) = 2 \geq 0$. *Among the three utilities,* $u_{0j}(t) = 0$ *is the largest for* $v_j \in (0, 2]$, $u_{2j}(t) = 4v_j - 8$ *is the largest for* $v_j \in [2, 12]$, *and* $u_{1j}(t) = 5v_j - 20$ *is the largest for* $v_j \in [12, \infty)$.

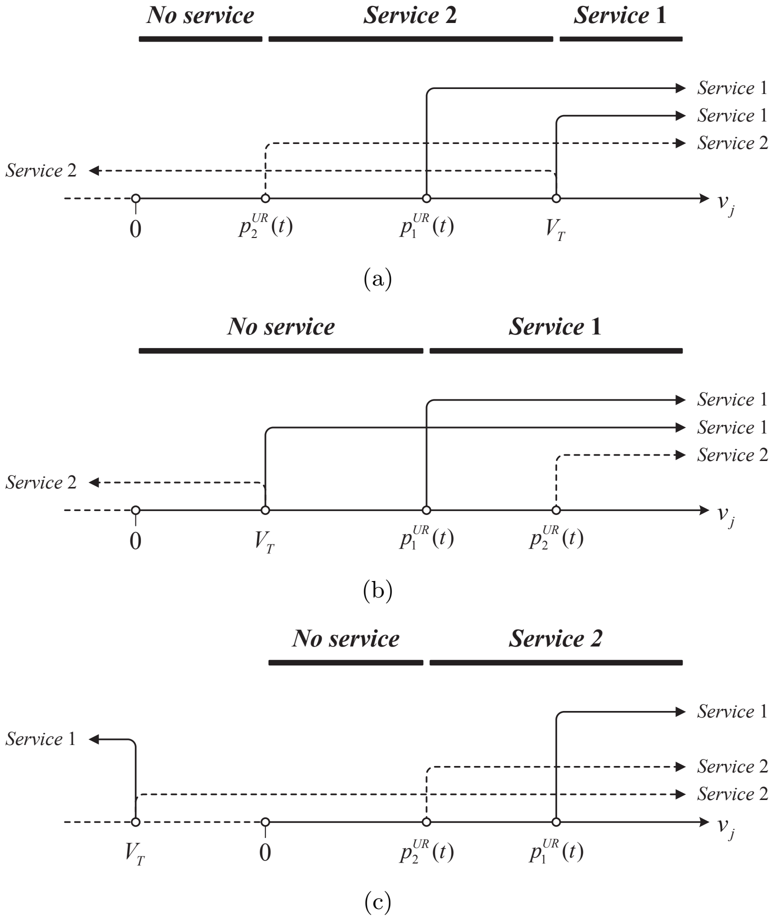


Figure 3.4: User subscription dynamics vs. valuation parameters of user j. (a) Case $\mathcal{A}$: $q_1(t-1) \geq q_2(t-1)$ and $p_1^{UR}(t) \geq p_2^{UR}(t)$, (b) Case $\mathcal{B}$: $q_1(t-1) \geq q_2(t-1)$ and $p_1^{UR}(t) < p_2^{UR}(t)$, and (c) Case $\mathcal{C}$: $q_1(t-1) < q_2(t-1)$ and $p_1^{UR}(t) \geq p_2^{UR}(t)$.

Case $\mathcal{B}$: $q_1(t-1) \geq q_2(t-1)$ and $p_1^{UR}(t) < p_2^{UR}(t)$

We now consider case $\mathcal{B}$ where the effective access rate of service 1 is greater than or equal to that of service 2, while the URP of service 1 is less than that of service 2, i.e., $q_1(t-1) \geq q_2(t-1)$ and $p_1^{UR}(t) < p_2^{UR}(t)$. In this case, the

SST V_T is less than the URP of service 1, i.e.,

$$p_2^{UR}(t) \geq p_1^{UR}(t) \geq V_T \geq 0. \tag{3.22}$$

Figure 3.4(b) illustrates case $\mathcal{B}$ in which there is no user subscribing to service 2. In other words, service 2 is less competitive than service 1 for any values of valuation parameter not only because of its high URP ($p_2^{UR}(t) > p_1^{UR}(t)$) but also because of its low effective access rate($q_2(t-1) \leq q_1(t-1)$). As a result, if $v_j \geq p_1^{UR}(t)$, user j subscribes to service 1; otherwise, user j does not subscribe to any services. Therefore, we have

$$g_0^{\mathcal{B}}(t) = F_V\left(p_1^{UR}(t)\right), \ g_1^{\mathcal{B}}(t) = 1 - F_V\left(p_1^{UR}(t)\right), \text{ and } g_2^{\mathcal{B}}(t) = 0. \tag{3.23}$$

Example 1-$\mathcal{B}$ *Consider a two-tier Internet access service in which* $p_1 = 12$*,* $p_2 = 8$*,* $q_1(t-1) = 6$*, and* $q_2(t-1) = 2$*. Then, the inequality in (3.22) is satisfied as follows:* $p_2^{UR}(t) = 4 \geq p_1^{UR}(t) = 2 \geq V_T = 1 \geq 0$*. Among the three utilities,* $u_{0j}(t) = 0$ *is the largest for* $v_j \in (0, 2]$*, and* $u_{1j}(t) = 6v_j - 12$ *is the largest for* $v_j \in [2, \infty)$*.*

Case $\mathcal{C}$: $q_1(t-1) < q_2(t-1)$ **and** $p_1^{UR}(t) \geq p_2^{UR}(t)$

We now consider case $\mathcal{C}$ where the effective access rate of service 1 is less than that of service 2, while the URP of service 1 is at least that of service 2, i.e., $q_1(t-1) < q_2(t-1)$ and $p_1^{UR}(t) \geq p_2^{UR}(t)$. In this case, the SST V_T becomes a negative value,

$$p_1^{UR}(t) \geq p_2^{UR}(t) \geq 0 \geq V_T. \tag{3.24}$$

Figure 3.4(c) illustrates case $\mathcal{C}$ in which there is no user subscribing to service 1. In other words, service 1 is less competitive than service 2 for any values of valuation parameter not only because of its high URP ($p_1^{UR}(t) \geq p_2^{UR}(t)$) but also because of its low effective access rate($q_1(t-1) < q_2(t-1)$). As a result, if $v_j \geq p_2^{UR}(t)$, user j subscribes to service 2; otherwise, user j does not subscribe to any services. Consequently, we have

$$g_0^{\mathcal{C}}(t) = F_V\left(p_2^{UR}(t)\right), \; g_1^{\mathcal{C}}(t) = 0, \text{ and } g_2^{\mathcal{C}}(t) = 1 - F_V\left(p_2^{UR}(t)\right). \tag{3.25}$$

Example 1-$\mathcal{C}$ *Consider a two-tier Internet access service in which* $p_1 = 10$, $p_2 = 8$, $q_1(t-1) = 2$, *and* $q_2(t-1) = 4$. *Then, the inequality in (3.24) is satisfied as follows:* $p_1^{UR}(t) = 5 \geq p_2^{UR}(t) = 2 \geq 0 \geq V_T = -1$. *Among the three utilities,* $u_{0j}(t) = 0$ *is the largest for* $v_j \in (0, 2]$, *and* $u_{2j}(t) = 4v_j - 8$ *is the largest for* $v_j \in [2, \infty)$.

Case $\mathcal{D}$: $q_1(t-1) < q_2(t-1)$ and $p_1^{UR}(t) < p_2^{UR}(t)$

Since $p_1 > p_2$ and $q_1(t-1) < q_2(t-1)$, the URP of service 1 satisfies the following inequality:

$$p_1^{UR}(t) = \frac{p_1}{q_1(t-1)} > \frac{p_2}{q_1(t-1)} > \frac{p_2}{q_2(t-1)} = p_2^{UR}(t). \tag{3.26}$$

However, this inequality contradicts with the given condition $p_1^{UR}(t) < p_2^{UR}(t)$. Therefore, case $\mathcal{D}$ cannot happen in the two-tier Internet access service.

Based on the subscription probabilities in (3.21), (3.23), and (3.25), the user subscription dynamics at time period t are represented by $n_0(t) = Ng_0^{(\cdot)}(t)$, $n_1(t) = Ng_1^{(\cdot)}(t)$ and $n_2(t) = Ng_2^{(\cdot)}(t)$.

3.5.2 Convergence of the User Subscription Dynamics

In this section, we examine the convergence condition of the user subscription dynamics in the two-tier service. In general, the case at time period t in section 3.5.1 can be different from the case at time $t-1$. In this situation, the subscription probabilities in (3.21), (3.23), and (3.25) may not stabilize to a fixed set of values. To ensure the convergence of such probabilities, we need to impose an additional requirement that the case does not change at the subsequent time point. Below, we will address this *retention* condition at two consecutive time points for each case in section 3.5.1.

Case $\mathcal{A}$: $q_1(t-1) \geq q_2(t-1)$ **and** $p_1^{UR}(t) \geq p_2^{UR}(t)$

Similarly to Lemma 1 for the non-tiered service, in this section, we can derive Lemma 4 which characterizes the condition on the access prices for the convergence of subscription probabilities in the two-tier service.

Lemma 4. *In a two-tier service where the capacities are C_1 and C_2, the access prices are p_1 and p_2 ($p_1 > p_2$), and the access rates are regulated to r_1 and r_2 ($r_1 > r_2$), both services always have a set of positive number of subscribers, if the following conditions are satisfied:*

$$p_1 \geq r_1 F_V^{-1}\left(1 - \frac{C_1}{Nr_2}\right) \text{ and } p_2 \geq r_2 F_V^{-1}\left(1 - \frac{p_1}{p_2}\frac{C_2}{Nr_1}\right).$$

Proof. In this proof, we derive a *strict* retention condition of case $\mathcal{A}$, i.e.,

$$q_1(t-1) \geq q_2(t-1) \Rightarrow \inf\left[q_1(t-1)\right] \geq \sup\left[q_2(t-1)\right], \tag{3.27}$$

and

$$p_1^{UR}(t) \geq p_2^{UR}(t) \Rightarrow \inf\left[\frac{p_1}{q_1(t-1)}\right] \geq \sup\left[\frac{p_2}{q_2(t-1)}\right], \tag{3.28}$$

where inf[·] and sup[·] are the *infimum* and the *supremum* of the corresponding parameters, respectively. Since $\sup[q_2(t-1)]$ is regulated to rate r_2, we can obtain the following equation by substituting (3.2) into (3.27):

$$\frac{C_1}{N \sup\left[g_1^{\mathcal{A}}(t-1)\right]} \geq r_2 \Rightarrow \sup\left[g_1^{\mathcal{A}}(t-1)\right] \leq \frac{C_1}{N r_2}. \tag{3.29}$$

Similarly, by substituting (3.2) into (3.27), we can derive

$$\frac{p_1}{r_1} \geq \frac{p_2 N}{C_2} \sup\left[g_2^{\mathcal{A}}(t-1)\right] \Rightarrow \sup\left[g_2^{\mathcal{A}}(t-1)\right] \leq \frac{p_1}{p_2}\frac{C_2}{N r_1}. \tag{3.30}$$

To derive the supremum of each subscription probability in all time periods, we investigate the boundary conditions of the subscription probabilities in (3.21). We first consider the boundary condition that $q_1(t-1) = q_2(t-1)$. In this condition, the value of SST V_T goes infinity which leads to zero subscriber to service 1. Thus, the subscription probability of service 2 satisfies the following inequality:

$$\begin{aligned} g_2^{\mathcal{A}}(t) &= 1 - F_V\left(p_2^{UR}(t)\right) \\ &\leq 1 - F_V\left(\min_{\forall t} p_2^{UR}(t)\right) \leq 1 - F_V\left(\frac{p_2}{r_2}\right) = \sup\left[g_2^{\mathcal{A}}(t)\right], \end{aligned} \tag{3.31}$$

where the first inequality of (3.31) holds because the CDF of valuation $F_V(v)$ is a non-decreasing function of the valuation parameter v, and the second inequality of (3.31) holds because the effective access rate of service 2 is regulated to rate r_2. Since the supremum is calculated over all time indexes in (3.31), we can obtain the following equation from (3.30):

$$1 - F_V\left(\frac{p_2}{r_2}\right) \leq \frac{p_1}{p_2}\frac{C_2}{N r_1} \Rightarrow p_2 \geq r_2 F_V^{-1}\left(1 - \frac{p_1}{p_2}\frac{C_2}{N r_1}\right). \tag{3.32}$$

Next, we examine the other boundary condition that $p_1^{UR}(t) = p_2^{UR}(t)$. Since $V_T = p_1^{UR}(t) = p_2^{UR}(t)$ in this condition, the subscription probability for service 1 is expressed in the similar manner:

$$\begin{aligned}g_1^{\mathcal{A}}(t) &= 1 - F_V\left(p_1^{UR}(t)\right)\\ &\leq 1 - F_V\left(\min_{\forall t} p_1^{UR}(t)\right) \leq 1 - F_V\left(\frac{p_1}{r_1}\right) = \sup\left[g_1^{\mathcal{A}}(t)\right].\end{aligned} \tag{3.33}$$

Similarly, by substituting (3.33) into (3.29), we can derive

$$1 - F_V\left(\frac{p_1}{r_1}\right) \leq \frac{C_1}{Nr_2} \Rightarrow p_1 \geq r_1 F_V^{-1}\left(1 - \frac{C_1}{Nr_2}\right). \tag{3.34}$$

From (3.32) and (3.34), we can prove this lemma. □

Case $\mathcal{B}$: $q_1(t-1) \geq q_2(t-1)$ and $p_1^{UR}(t) < p_2^{UR}(t)$

As we stated in (3.23), if case $\mathcal{B}$ happen at time $t-1$, no user subscribes to service 2 at time t. Then, it is probable that a group of users choose service 2 at time $t+1$, because $q_1(t)$ is less than or at least equal to $q_1(t-1)$ while $q_2(t)$ has maximum value. Here, in order for both services to have a set of positive steady-state number of subscribers, case $\mathcal{B}$ should be followed by case $\mathcal{A}$ at time t, and then, the case $\mathcal{A}$ should be kept over time. However, this scenario is obviously a subset of the retention condition of case $\mathcal{A}$, which specifies the condition under which case $\mathcal{A}$ does not change at the subsequent time point, so we need not consider this scenario.

Meanwhile, the retention condition of case $\mathcal{B}$, which is the condition that case $\mathcal{B}$ does not change over time, can be seen as an example of the non-tiered service where only service 1 is operated. Thus, we do not take into account case $\mathcal{B}$ in the tiered service.

Case $\mathcal{C}$: $q_1(t-1) < q_2(t-1)$ and $p_1^{UR}(t) \geq p_2^{UR}(t)$

Similarly, we need only consider the retention condition of case $\mathcal{C}$. According to (3.25), there is no user who subscribe to service 1 in case $\mathcal{C}$, which can be also seen as an example of non-tiered service where only service 2 is operated. Thus, we do not consider case $\mathcal{C}$ in the tiered service as well.

3.5.3 Optimal Pricing for Maximizing the ISP Revenue

We now discuss the revenue maximization for the tiered service. From (3.6), the total ISP revenue for two services is given by

$$R = \mathbf{n}^* \cdot \mathbf{p} = n_1^* p_1 + n_2^* p_2. \tag{3.35}$$

In (3.12) and (3.13), the steady-state number of subscribers for each service can be written in two different forms according to its value as in the non-tiered service. Then, the set of the steady-state numbers of subscribers for two services is defined under four combinations of conditions, which makes the analysis more complicated. However, since the maximum ISP revenue for each service occurs in the *regulated range* as proved in Lemma 3, we can focus only on the case where the access rates for both services are regulated. Thus, we write the steady-state numbers of subscribers for two services as follows:

$$\begin{aligned} n_1^* &= N\left(1 - F_V\left(\frac{p_1 - p_2}{r_1 - r_2}\right)\right) \\ n_2^* &= N\left(F_V\left(\frac{p_1 - p_2}{r_1 - r_2}\right) - F_V\left(\frac{p_2}{r_2}\right)\right) \end{aligned} \tag{3.36}$$

As a result, we formulate the revenue maximization problem for two-tier service as follows:

Problem 2.

maximize $\quad N\left(p_1 - (p_1 - p_2)\, F_V\left(\frac{p_1 - p_2}{r_1 - r_2}\right) - p_2 F_V\left(\frac{p_2}{r_2}\right)\right)$

subject to

$$p_1 - p_2 \geq (r_1 - r_2) F_V^{-1}\left(1 - \frac{C_1}{N r_1}\right),$$

$$p_1 \geq r_1 F_v^{-1}\left(1 - \frac{C_1}{N r_2}\right),$$

$$p_2 \geq r_2 F_V^{-1}\left(1 - \frac{1}{N} min\left(\frac{p_1}{p_2}\frac{C_2}{r_1}, \frac{C_1}{r_1} + \frac{C_2}{r_2}\right)\right),$$

$$C_1 + C_2 = C$$

variables p_1, p_2, C_1, C_2

Note that the capacities for two service tiers as well as the access prices are used as decision variables in the revenue maximization problem for the tiered service. However, as can be seen in the problem, they do not directly affect the total ISP revenue, but rather determine the boundary values of the price conditions. We will discuss the impacts of the capacity partitioning in more detail with some numerical examples in the following section.

We can also apply the general problem to the particular case where the valuation parameter follows a truncated normal distribution in (3.3)

Problem 2-1.

maximize $\frac{N}{2}\left(p_1 - (p_1 - p_2)\, erf\left(\frac{\frac{p_1 - p_2}{r_1 - r_2} - \mu}{\sigma\sqrt{2}}\right) - p_2 erf\left(\frac{\frac{p_2}{r_2} - \mu}{\sigma\sqrt{2}}\right)\right)$

subject to

$$p_1 - p_2 \geq (r_1 - r_2)\left(\mu + \sigma\sqrt{2}\ erf^{-1}\left(1 - \frac{2C_1}{N r_1}\right)\right),$$

$$p_1 \geq r_1\left(\mu + \sigma\sqrt{2}\ erf^{-1}\left(1 - \frac{2C_1}{N r_2}\right)\right),$$

$$p_2 \geq r_2\left(\mu + \sigma\sqrt{2}\ erf^{-1}\left(1 - \frac{2}{N} min\left(\frac{p_1}{p_2}\frac{C_2}{r_1}, \frac{C_1}{r_1} + \frac{C_2}{r_2}\right)\right)\right),$$

$$C_1 + C_2 = C$$

variables p_1, p_2, C_1, C_2

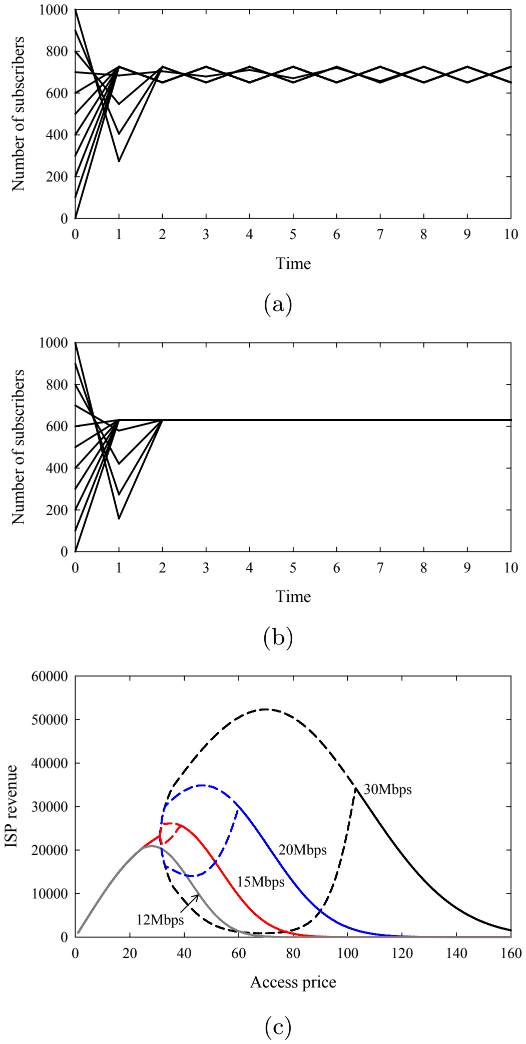
As in the case of the non-tiered service, we can rewrite the problem in a polynomial form by expanding the error function and its inverse function into their Taylor series, and compute the approximations of the maximum revenue and the optimal access prices from the low-order terms of this problem. In particular, the first-order approximation converts the problem to a quadratic program, which can be easily solved by conventional methods. Even though the higher-order approximation is used for more accuracy, the computation time of the naïve method is acceptable. In this Chapter, we omit the computational details of this method for the expressional brevity.

3.6 Numerical Results

We now present numerical examples which illustrate the analysis of optimal pricing for non-tiered and tiered services. In the examples, we consider 10 Gbps total capacity and 1000 potential users. The valuation parameter of each user is assumed to follow a truncated normal distribution with mean $\mu = 3$ and standard deviation $\sigma = 1$. We use MATLAB to emulate the user subscription dynamics and to solve the revenue maximization problem.

3.6.1 Non-tiered Service Example

First, we illustrate the user subscription dynamics for the non-tiered service in Figures 3.5(a) and 3.5(b). We consider the regulation rate of 15 Mbps and set the access prices to \$ 36 and \$ 40, respectively. The results for these two prices are completely different from each other as shown in the figures. At the price of \$ 40, the number of subscribers converges to a steady-state value as time proceeds, regardless of the initial value, but at \$ 36, the number



(a)

(b)

(c)

Figure 3.5: Results of non-tiered service examples. (a) and (b) show the examples of the oscillation and convergence of user subscription dynamics, respectively; (c) shows the variation of ISP revenue according to access price for different regulation rates.

of subscribers oscillates continually between two subscription values. This difference is essentially originated from the property of the effective access rate that it is divided into two distinct ranges.

Figure 3.5(c) shows the variation of ISP revenue according to access price for different regulation rates in the non-tiered service. The dashed lines for each regulation rate represent the upper and lower values of the ISP revenue when the user subscription dynamics oscillates between them. On the left side of the oscillation range (i.e., the access price is low), the number of subscribers converges to the steady-state value which belongs to the *decreasing range*; on the right side, the steady-state number of subscribers occurs in the *regulated range*. In this figure, for the low regulation rate (e.g., 12 Mbps), the ISP revenue always converges regardless of the access price. In this case, the maximum ISP revenue occurs at the vertex (i.e., $p = \hat{p}$). For higher regulation rates, on the other hand, the ISP revenue has the maximum value at the right edge price of the oscillation range (i.e., $p = p_h$).[10] We can obtain the optimal price for each regulation rate by solving Problem 1-1, or simply by using the equations (3.14) and (3.16).

3.6.2 Tiered Service Example

Next, let us consider the examples for the tiered service. In these examples, for simplicity, we consider two-tier service over dually-partitioned VNs. Figures 3.6(a) and 3.6(b) describe the user subscription dynamics under two different price vectors: $\mathbf{p} = [0, 70, 27]$ and $\mathbf{p}' = [0, 70, 35]$. We assume that, in both examples, the access rates are regulated to $r_1 = 20$ Mbps and $r_2 = 10$ Mbps, and the partitioning coefficients are set to $\theta_1 = 0.6$ and $\theta_2 = 0.4$ for two service tiers.

---

[10] Note that the ISP revenue at the right edge price of the oscillation range is always larger than that at the left edge price as proved in Lemma 3.

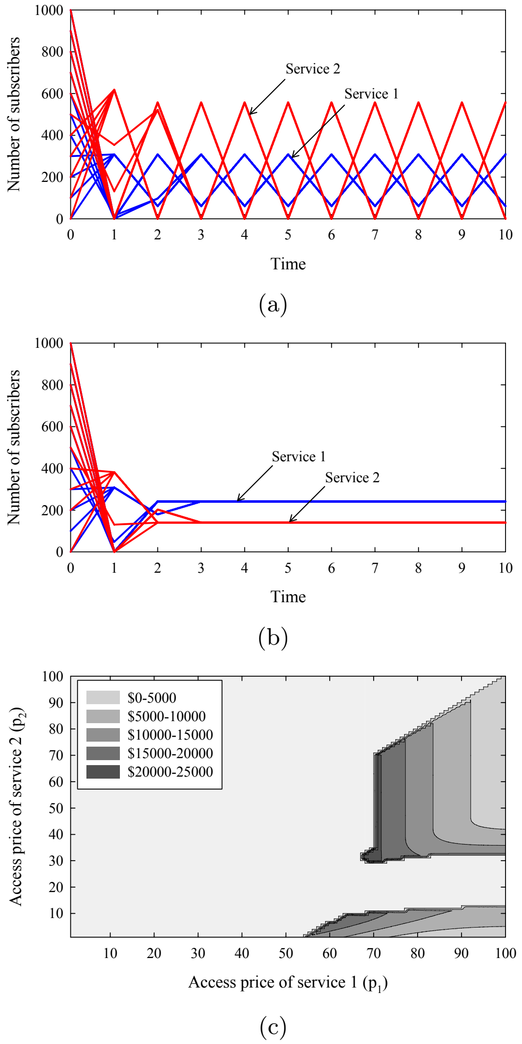
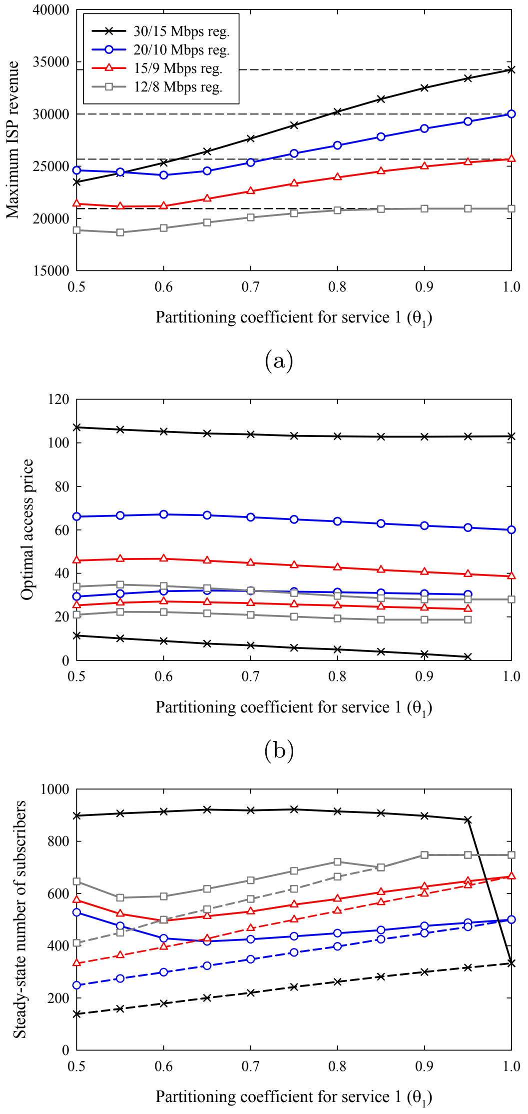


Figure 3.6: Results of tiered service examples. (a) and (b) show the user subscription dynamics at two different price vectors **p** and **p′**, respectively; (c) shows the variation of ISP revenue according to the price setting for two services.

In Figure 3.6(a), the numbers of subscribers for both services oscillate. Although the price setting in this example does not meet the oscillation condition specified in Lemma 1, it satisfies the conditions of Lemma 4. That is, since the values of user's valuation for two services are turned over continuously at this price setting, some users repeatedly switch over to the other service at every time period. In contrast, in Figure 3.6(b), the user subscription dynamics is stabilized to the equilibrium state after a few periods.

We present the variation of the ISP revenue according to the price setting for two services as a contour plot in Figure 3.6(c). Other parameters except the prices are the same as in the upper figures. The brightest area in this figure indicates the condition under which the ISP revenue does not have a steady-state value due to the oscillation of the user subscription dynamics. Since the price condition for convergence in the tiered service is much stricter than that of the non-tiered service, the price condition that generates *real* revenue occupies little area in the figure. Thus, pricing for tiered service needs to be carried out with careful consideration in order to obtain stable revenue. The optimal set of prices that maximizes the ISP revenue can be obtained by solving Problem 2-1.

We now discuss the effect of capacity partitioning on the ISP revenue in the tiered service. Figure 3.7(a) shows the maximum ISP revenue according to the partition ratio for several different regulation rates. We consider four combinations of regulation rates: 30 and 15 Mbps; 20 and 10 Mbps; 15 and 9 Mbps; and 12 and 8 Mbps for service 1 and service 2, respectively. In the figure, the horizontal dashed lines represent the values of the maximum ISP revenue for the non-tiered service when their regulation rates are 30, 20, 15, and 12 Mbps, from the top, respectively. As the partitioning coefficient of service 1 θ_1 increases, the maximum ISP revenue for each combination grows, and then, at $\theta_1 = 1$ (i.e., in the case that service 1 monopolizes the resources), it reaches the



(a)

(b)

(c)

Figure 3.7: (a) Maximum ISP revenue, (b) optimal prices, and (c) steady-state number of subscribers according to partition ratio for different regulation rates.

maximum value, which is the same as the value of the corresponding dashed line. This result indicates that the tiered service cannot be more profitable than the non-tiered service unless the highest regulation rate of the former is larger than the regulation rate of the latter. On the contrary, however, if the highest regulation rate of the tiered service is set to a larger value than the regulation rate of the non-tiered service (e.g., 15 and 9 Mbps vs. 12 Mbps), the tiered service can generate more revenue than the non-tiered service along with a proper capacity partitioning.

Finally, we investigate the interaction between capacity partitioning, pricing, and the number of subscribers. Figures 3.7(b) and 3.7(c) describe the optimal prices for two services and the steady-state numbers of subscribers at that time, respectively, according to the partition ratio for different combinations of regulation rates. In Figure 3.7(b), for each combination, two lines are depicted: The upper line represents the optimal access price for service 1 and the lower for service 2. In Figure 3.7(c), on the other hand, two lines refer to the total number of subscribers (solid line) and the number of subscribers for service 1 (dashed line), respectively; that is, the gap is the number of subscribers for service 2. These two figures suggest that capacity partitioning has a more decisive effect on the steady-state number of subscribers than on the optimal pricing; in fact, the optimal pricing is hardly affected by the capacity partitioning, but rather depends crucially on the regulation rates. This result stems from the fact that the capacity increase leads to a raising of the threshold number of subscribers (i.e., C/r) from which the access rate drops, not to an increase in maximum access rate of each user. Since the steady-state number of subscribers, which maximizes the ISP revenue, is determined around this threshold value (i.e., at $p = p_h$), the capacity increase for a service induces an increase in the number of subscribers for the service. Thus, by using the capacity partitioning, ISPs can efficiently estimate and determine the proportion of the number of subscribers for their service tiers.

3.7 Related Work and Discussion

In this section, we review some related studies in the fields of both networking and economics, and highlight several contributions of this study in comparison with them.

Network virtualization is an emerging technology that enables dynamic partitioning of a shared physical network infrastructure into multiple VNs. Because of the flexibility in resource allocation and independency among VNs, it has been mainly deployed in building a testbed such as PlanetLab [1] and GENI [21]. Accordingly, most of the early studies [6–9] and even several recent studies [10–13] on VNs have focused on the on-demand embedding of each VN request onto the remaining substrate network resources. However, since several *pluralist* studies suggested that virtualization should become a fundamental attribute of the Internet [3–5], network virtualization has also come to be regarded as a means of diversifying the Internet, not just a tool for experimentation. In this context, network virtualization still faces a number of challenging issues that need to be addressed, including admission control, addressing, failure handling, security and privacy, and network economics [14].

Among them, the economic issues of VNs have rarely been dealt with in the literature. A study on service overlay network [25] emphasized the balance between bandwidth costs and penalties due to QoS violation in the bandwidth provisioning problem for a service overlay network. Several recent studies on VN embedding focused on cost-effective resource allocation by introducing the price of specificity which captures the resource cost of the embedding under a given specification [12], and considering the migration cost of already embedded VN elements to new locations [13]. Another study in [54] analyzed the economic incentives of ISPs for supporting the global availability of a virtualized testbed, taking into account three different market types. These studies, however, either simply use economic variables such as ISP revenue, bandwidth cost, and penalty in order to assess the efficiency of the resource allocation in VNs, or focus only on the economics of testbed itself, rather than provide a new economic model based on (or using) VNs. Thus, our approach has a significant contribution in the sense that we employ VNs as a solution to improve the existing economic model.

Meanwhile, in the field of economics, capacity partitioning for supporting tiered service is, in fact, not a new concept. *Paris Metro pricing* (PMP) also assumes a set of logical subnetworks [43], albeit not specifying VNs.[11] Compared with the existing studies on tiered pricing including PMP, our approach can be characterized in technical terms as follows: We consider capacity sharing service [48, 49, 51], assume users' opt-out as well [45, 46], and employ a general distribution of the users' valuations of the QoS [46]. Without considering the access rate regulation and its effects on the user subscription dynamics and ISP revenue, our results are in general accord with those of several studies on PMP [49, 51] under certain conditions. However, since the PMP stems from the fundamental principle that differentiated pricing generates the service level differentiation, it does not regard the capacity of each subnetwork as a dynamically-controllable variable, which precludes individual

---

[11] In this respect, our approach could be viewed as an extension of the PMP scheme.

QoS guarantees. In our model, on the other hand, price, capacity, and even regulated rate for each VN are under the control of ISP. Thus, our approach can provide more fine-grained QoS management than PMP scheme.

Finally, in contrast with other studies, our model captures the regulation of each user's access rate by the operational policy of the ISP or the performance limitation of hardware. By applying the access rate regulation to the analysis, we can obtain several interesting results as follows:

- The access rate regulation directly affects the convergency of user subscription dynamics in both non-tiered and tiered services. Under certain price conditions, the number of subscribers for each service is not stabilized to a steady-state point over time, but oscillates indefinitely around it.

- The maximum ISP revenue always arises when the access rate of each user has the maximum value in both non-tiered and tiered services. That is, by optimal pricing, ISPs can provide users with QoS guarantees, which is contrary to the PMP.

- In any case, adding a lower service level to the tiered structure generates lower revenue than doing nothing. On the other hand, if suitable regulation rates are applied, the tiered service can be more profitable than the non-tiered service.

The above observations are expected to provide useful guidelines for deploying a tiered Internet access service.

3.8 Summary

In summary, we analyzed the economical efficiency of tiered access service offered over a layered network architecture based on virtualization technology. Under the assumption of access rate regulation, we examined the user subscription behavior and found the optimal pricing and capacity partitioning to maximize the ISP revenue in tiered access service. We verified that the tiered service can make more profits than the non-tiered service along with the proper price structure and capacity. We believe that, not only does our analytical study on the VN-based tiered service provide ISPs with a more practical guideline for pricing and capacity partitioning of tiered access service, but it also enables to overcome the structural limitations of the existing PMP scheme.

Finally, we discuss several limitations of our study and the future directions. First, we adopt the single capacity network model to simply indentify the relation between the effective access rate of each user and the number of users, so our result, i.e., optimal capacity partitioning coefficient, does not indicate the actual partitioning ratio for all links in the network. To operate tiered access service based on VNs in practice, capacity dimensioning and partitioning for each link in the network should be further performed with consideration for other variables including the network topology and user distribution. Next, we focus primarily on the effect of the access rate regulation on the user subscription dynamics, but do not address the possibility that it directly affects the average traffic usage of each user. Further attention could be paid to capturing the intuition that the higher the regulation rate, the more traffic the user is probably forced to generate. To this end, we are currently working on the development of an extended analytical model for the access rate regulation. Lastly, in this study, we only consider a static virtualization scenario in which the number of VNs and the proportion of capacity partitioning for each VN are not changed over time. Developing a dynamic capacity partitioning method to maximize the ISP revenue can be another interesting research topic for future study.

Chapter 4

Inter-Virtual Network Traffic Switching for Fast Failure Recovery

4.1 Introduction

Network virtualization is an emerging technology that enables the dynamic partitioning of a shared physical network infrastructure into multiple virtual networks (VNs). Because of its flexibility in resource allocation and independency among VNs, network virtualization has been mainly deployed to build a testbed network (e.g., PlanetLab [1], VINI [21], and GENI [2]). Recently, network virtualization has even come to be regarded as a cost-effective solution for diversifying the Internet, not just a tool for experimentation [3–5]. Service providers can easily create their own customized (virtual) networks with more reasonable costs by sharing physical resources with others, instead of leasing expensive dedicated lines and ports from infrastructure providers, which will allow the Internet to support more diverse services, offered by various types of providers.

Over the last decade, numerous studies have been devoted to the technical and theoretical issues related to the network virtualization. The majority of these studies have focused on the efficient embedding of virtual resources (i.e., virtual nodes and links) onto the underlying physical resources since it is of practical importance in operating and managing the testbed [6–11]. As a means of building the *layered* Internet, however, network virtualization still faces a number of challenging issues that need to be addressed, including admission control, addressing, failure handling, security and privacy, and network economics [14].

Among them, one of the most critical issues of the network virtualization is failure recovery. Because of the sharing of network resources with other VNs and the instability of virtualization itself, VNs are more prone to suffer from wide variety of failures than a single-layered network. Leaving aside the hardware failures that affect all the VNs hosted on the failed components, it is meaningful and necessary to develop an effective recovery mechanism for a single VN to cope with simple node and link failures occurred within the VN. Although the existing mechanisms, such as IP fast reroute [55], provide protection against such failures within a very short time in a single-layered network, the pre-computation of the backup paths and the presence of topological constraints restrict the application of these approaches to VN.[1]

In a network virtualization environment (NVE), multiple VNs coexist in parallel. Accordingly, between arbitrary two physical nodes, there always exist multiple paths via different VNs, each of which runs an independent routing protocol. This structural feature of NVE can be exploited in determining the

---

[1] We discuss the details of these mechanisms and their limitations in the following section.

backup path for each failed node or link. Once a failure occurs, if the adjacent virtual nodes immediately switch the interrupted traffic to other VN which has valid routes towards the opposite ends of the failed component, the duration of traffic forwarding disruption caused by the failure will be greatly reduced. On the basis of this concept of inter-VN traffic switching, in this Chapter, we develop a fast and effective failure recovery mechanism for an NVE.[2]

The failure recovery through inter-VN traffic switching neither has topological constraints for the existence of backup paths, nor requires the precomputation of them, but it guarantees as fast recovery as the existing mechanisms. Since detouring via other VN is performed in accordance with the routing table of that VN, each virtual node need not be involved in the determination of detour paths for its outgoing interfaces or adjacent nodes, but only have to know the end point of each detour path. Despite these advantages, however, the proposed failure recovery mechanism has a potential risk that the switched traffic could cause QoS degradation in the neighbor VN. Thus, throughout this Chapter, we pay careful attention to the analysis of this possible adverse effect and the design of a practical inter-VN traffic switching method that can minimize the QoS degradation.

The rest of this Chapter is organized as follows: Section 4.2 discusses the characteristics of failures in the Internet and related work on fast failure recovery. Section 4.3 gives some preliminary considerations, including VN model, design goals and business models. In Sections 4.4, we describe the inter-VN traffic switching mechanism and failure recovery process based on it. In Section 4.5, we present a numerical analysis of the effect of traffic switching

---

[2] This work was conducted with Prof. Sueng-Yong Park and Prof. Seung-Woo Seo.

on the QoS in the neighbor VN. Finally, we conclude in section 4.6 with further discussion.

4.2 Background

In today's Internet, failures occur for various reasons, including an accidental fiber cut, linecard failure, router crash or reboot, router processor overload, software error, and misconfiguration. Among them, apart from failures due to scheduled network maintenance, around 70% of the unplanned failures are individual link failures, and half of the remaining failures are node failures affecting all connected links [56].

In the case of an NVE, although no detailed analysis has been reported, more diverse types of failure events could happen even more frequently. First of all, the partitioning of physical resources simply increases potential sources of failure. Moreover, the layered structure shared by multiple VNs and the attendant complexity are likely to increase the possibilities of router crashes, errors, and misconfigurations. The instability of router virtualization itself may also be a significant cause of failure. Thus, failure recovery has become more important in NVE.

Conventional routing protocols such as OSPF and ISIS handle failures in several steps: failure detection, failure dissemination, and routing table re-computation. However, this routing re-convergence process takes up to several hundred seconds depending upon the number of network prefixes, router architecture, and processor speeds [57], which is too large for applications with real time demands [58]. A fine-tuning of detection parameters can accelerate the re-convergence process, but too rapid and frequent advertisements of failure information can increase routing instability [59].

The IETF has defined a framework called IP Fast Reroute (IPFRR) [55] in order to ensure failure recovery within tens of milliseconds. IPFRR reduces the failure reaction time through local rerouting and pre-computed detours. Once a failure occurs, the routers directly adjacent to the failure immediately forward the interrupted traffic to pre-computed alternate next-hops, called loop-free alternates (LFAs) [60]. Since LFAs do not provide full coverage of both link and node failures, the IETF has recently drafted a tunneling approach using *not-via* addresses [61]. Besides these standardized methods, many other IPFRR approaches have been proposed, some of which attempt to change IP's destination-based forwarding [62,63], and some others use extra bits in the IP header [64, 65], in order to explicitly distinguish the detouring packets from the other normal packets traveling on their shortest paths. Meanwhile, MPLS fast reroute protects a label-switched path (LSP) from single node and link failures, which attains the same 50-ms recovery as SONET [66].

Even though IPFRR is straightforward and unobtrusive, it has a significant problem that not all routers have backup paths (i.e., LFAs) to all other routers. Recent studies have pointed out this topological constraint [67], and showed that cleverly adding a few new links can improve the protection coverage [68]. From the point of view of service providers, however, leasing extra dedicated lines and ports just for backup paths is wasteful and inefficient, even in NVE.

Meanwhile, there have been several studies, albeit few in number, attempting to design resilient network architecture based on virtualization. Resilient overlay network (RON) [69] allows distributed Internet applications to detect and recover from failures and periods of degraded performance within several seconds, by periodically monitoring the functioning and quality of the overlay paths at every node, and using this information to decide whether to route packets directly over the Internet or by way of other RON nodes. In DaVinci architecture [70], each substrate link periodically reassigns bandwidth shares between its virtual links, while at a smaller timescale, each VN runs a distributed protocol that maximizes its own performance objective independently. Through such adaptive resource allocation between VNs, the DaVinci architecture can dynamically recover from failures which occur within each VN. However, these approaches commonly require the deployment of particular network architecture with its own monitoring functionality, which certainly restricts their practical applications.

In this study, we aim to develop a more general and legacy-compatible failure recovery mechanism for an NVE. The proposed recovery mechanism based on inter-VN traffic switching is expected to provide as fast recovery as IPFRR methods from diverse types of node and link failures occurring within a single VN.

4.3 Preliminaries

4.3.1 Virtual Network Model

In this study, we consider a general form of VN, which is a group of virtual nodes connected together via a set of virtual links. As described in Figure 4.1, each VN is essentially built as a subset of the underlying substrate network topology. In a VN, each virtual node is a logical partition of the physical resources, hosted on a virtual router (VR); each virtual link is (in general)

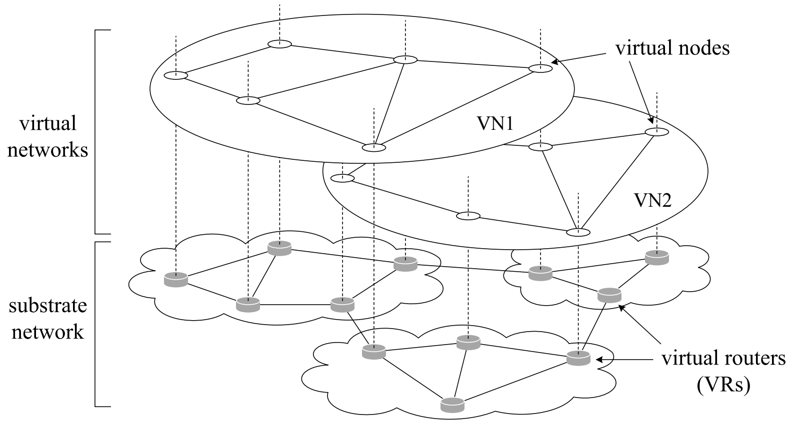


Figure 4.1: Network virtualization environment.

a logical tunnel (i.e., IP-in-IP, IPsec, MPLS) over one or multiple physical links between two virtual nodes, occupying a certain portion of bandwidth resources.

In an NVE, the roles of the traditional Internet service providers (ISPs) are decoupled into two independent participants: infrastructure providers (InPs) and service providers (SPs) [4, 5]. InPs manage the physical resources of the substrate network, and offer their resources to different SPs through programmable interfaces. SPs lease physical resources from one or more InPs to create VNs, and provide their end users with customized services on the VNs. Each SP establishes service level agreements (SLAs) with the underlying InPs to stipulate the level of connectivity, performance, and other constraints.

In this Chapter, for clarity of exposition, we make the additional simplifying assumptions on the VN model.

First, each VN is completely isolated from the other coexisting VNs. Thus, any failure in one VN is contained within itself and leaves all the other VNs unaffected.[3] Besides, the traffic congestion occurring in a VN does not affect the performance of the others.

Next, every VR is shared by more than one VN. That is, every virtual node has an access to at least one other failure-free VN hosted on the same VR. Thus, every link of each VN has more than one disjoint backup link or path (i.e., a series of links) on other VN. This does not mean that every VN has the same topology.

Finally, every VN offers connectivity to all the end nodes. If some packets are switched to other VN on their way to destinations in order to detour around a failed node or link, they should be able to reach the desired destinations.

We now define some terms used in this Chapter. Two VNs involved in traffic switching process are in *neighbor* relationship.[4] The physical point where one virtual node switches traffic to the neighbor virtual node hosted on the same VR is called *switching point*; similarly, the point where the switched traffic is returned from the neighbor VN to its original VN is referred to as *returning point.*

4.3.2 Design Goals

In this study, we aim to design an effective and efficient failure recovery method through inter-VN traffic switching in NVE. In order that the Inter-VN traffic switching would be considered as a more attractive alternative for failure

---

[3] In this study, we confine our consideration to the software failures which do not affect other VNs sharing the same components, such as buffer overflows, software errors, and protocol misconfigurations.

[4] The term *neighbor* is also used to indicate the nodes which run the same routing protocol. To avoid confusion, we use the term *neighbor* together with VN or virtual node when referring to the switching opponent.

recovery than the existing fast rerouting methods used within a single VN, it must fulfill the following design criteria:

1. Fast Recovery

IP fast reroute employs local rerouting in which only the adjacent nodes are notified of the failure, which leads to a significant reduction of the service disruption time by eliminating a time-consuming routing re-convergence process. The adjacent nodes experience forwarding discontinuity only during the detection period. This is theoretically the fastest recovery time without improving the failure detection. Thus, in order to provide as fast failure recovery as IP fast reroute, we similarly use local switching where the interrupted traffic is switched to the pre-contracted neighbor VN directly at the virtual nodes that detect the failure.

2. QoS guarantee

Packets switched from a VN are forwarded over the neighbor VN until the failure is corrected or the alternative paths are founded within their original VN. Here, in case that the neighbor VN provides its own service to the end users,[5] some of them will unavoidably suffer a certain degree of QoS degradation due to the unexpected traffic inflow from other VN. Thus, we need to carefully analyze the impact of traffic switching on the QoS of the neighbor VN and design the switching method to ensure the minimum QoS degradation.

3. Full protection

The failure detection mechanism of routing protocols, which uses *dead* period, does not notify the root cause of failures. Thus, the failure recovery

---

[5] we discuss the business models for inter-VN traffic switching in the next subsection.

methods should be able to cope with both node and link failures. In the inter-VN traffic switching, the number of hops that the switched traffic travels over the neighbor VN determines the protection coverage against failures. In order to provide full protection against diverse types of failures, we need to allow the switched traffic to travel as many hops over the neighbor VN as possible. Higher utilization of network resources in the neighbor VN, however, leads to more QoS degradation. Considering this trade-off, we present three options of returning strategies for inter-VN traffic switching.

4. Compatibility

The NVE basically allows heterogeneity of VNs. Thus, the switched packets need to be encapsulated in an appropriate tunneling protocol (e.g., IP-in-IP, IPsec, MPLS) in order to be delivered over the neighbor VN, which may have a different address structure, or even use different underlying protocols. Moreover, the inter-VN traffic switching should work well with the failure detection and routing re-convergence processes of the underlying routing protocol.

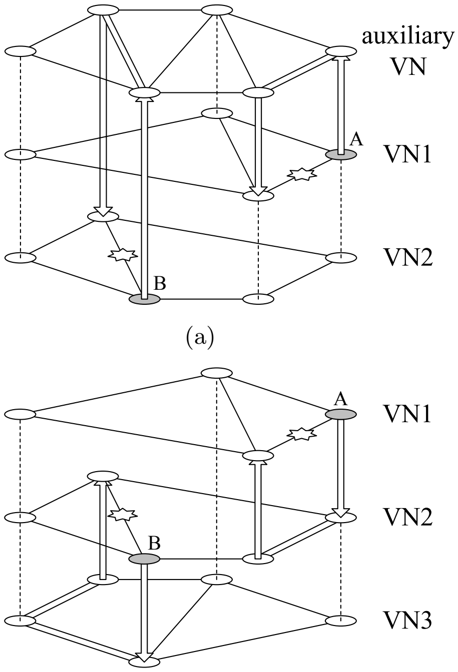
4.3.3 Business Models and Switching Policy Agreement

In this section, we present two types of business models for the NVE supporting inter-VN traffic switching: auxiliary and cooperation models. Figs. 4.2(a) and 4.2(b) show examples of these business models.[6]

In the former model, each InP operates an auxiliary VN, which offers the alternative paths to all VNs hosted by the InP in the presence of failures. In Figure 4.2(a), for example, both node A of VN 1 and node B of VN 2

---

[6] In both figures, for ease of understanding, we consider only one-directional traffic switching and simple next-hop returning strategy, which is discussed below.



(a)

(b)

Figure 4.2: Business models: (a) auxiliary model and (b) cooperation model.

switch the interrupted traffic to the auxiliary VN to avoid failures. In this business model, since a certain portion of the physical resources are reserved for placing the auxiliary VN, each InP should accept the corresponding loss. The improved availability of VNs due to the auxiliary VN, on the other hand, will attract more SPs, which may cover up this loss. The auxiliary model is quite straightforward and does not call for a detailed analysis. Thus, in this study, we do not focus on this model, but rather make a careful analysis of the failure recovery process based on the following model.

The cooperation model is characterized by the interaction between multiple VNs. In this business model, two SPs make a contract to utilize each other's VN in case of failures. That is, once a failure occurs on a VN, the interrupted traffic is immediately switched into the pre-contracted neighbor VN in order to detour around the failure. In Figure 4.2(b), node A of VN 1 switches traffic to pre-contracted VN 2, and so does node B of VN 2 to VN 3. From the point of view of SPs, since they can make their VNs more reliable and resilient to failures without an additional purchase of physical resources for backup paths, the cooperation with other VNs is clearly an attractive and economical solution. Although VNs may suffer QoS degradation due to the unexpected traffic inflow from the neighbor VNs, not only does this adverse effect appear during a short period, but also it can be alleviated by optimizing the switching parameters.

In the cooperation model, one VN switches packets to the neighbor VN on the basis of a switching policy agreement (SPA). The SPA specifies the switching requirements in terms of returning strategy for the switched packets and peak switching rate.

We first discuss the returning strategy, which is the most critical element of the SPA. The virtual node which detects a failure encapsulates the following packets with an additional header, and forwards them over the neighbor VN. Here, the destination address of the tunneling header indicates the returning point. To determine where to return the switched packets is closely related to the protection coverage against different types of failures and the QoS of the neighbor VN. We consider three types of returning strategies according to returning point as follows.

Next-hop return – The returning point is set to the next-hop node beyond the failure. This strategy can cope only with a single link failure. However, although the number of hops along which the switched packets actually travel over the neighbor VN may be more than one depending on the corresponding shortest path routing, the next-hop return intuitively guarantees the minimum QoS degradation of the neighbor VN in comparison with the other strategies.

Next-next-hop return – The returning point for each of the switched packets is two hops away along its original path. Since the interrupted packets may have different (final) destinations, the next-next-hop address differs from packet to packet. However, router in general has no information about its two-hop neighbors. Thus, each virtual node needs to pre-compute the next-next-hop addresses for all the entries in its routing table with an additional algorithm.[7] The next-next-hop return provides protection against both node and link failures, and its effect on the QoS degradation in the neighbor VN will not be significant, albeit larger than the effect of the simple next-hop return.

No return – The switched packets are not returned and forwarded up to their final destinations over the neighbor VN. Since this strategy causes a high consumption of network resources in the neighbor VN, it will not be a good option in terms of QoS degradation. However, it provides full protection from wide-ranging network outages as well as both node and link failures.

Table 4.1 summarizes the characteristics of the above three types of returning strategies. Each SP stipulates one of these returning strategies or the combination of different strategies for its nodes into the SPA, taking into account its service utilization pattern, network topology, and frequently occurring failure types. For example, if a virtual node often crashes due to stack overflow, the SP can use the next-next-hop return strategy at the adjacent nodes to that node while applying the next-hop return to the other nodes.

---

[7] Not-via approach [61] provides a method to compute the next-next-hop address.

Table 4.1: Characteristics of Returning Strategies

Returning strategy	*Next-hop return*	*Next-next-hop return*	*No return*
Protection coverage	link failure	node / link failures	wide-ranging failures
QoS degradation	minimum	low	high
Pre-configuration	no	yes	no
Differentiation	per-next-hop	per-destination	not necessary
Encryption	yes	yes	optional

We now discuss the peak switching rate. The inflow of a burst of packets from the neighbor VN can cause network congestion, which will have an adverse effect on QoS. Thus, each SP allows its virtual nodes to regulate the peak traffic inflow rate from the neighbor VN in order not to violate SLAs with its end users. Each SP can configure different regulation rates at its virtual nodes in the light of link utilization. The details of the inflow rate regulation process and its impact on the QoS degradation will be investigated in the following sections.

4.3.4 Other Considerations

Besides the functional requirements described above, several other considerations should also be taken into account in designing the failure recovery mechanism based on inter-VN traffic switching.

First, we present the addressing issue. Network virtualization can be categorized according to the virtualized layer of the network stack, starting from the physical layer (e.g., UCLP [71]) and continuing up to the application layer (e.g., PlanetLab [1] and VIOLIN [72]). Since the virtualized layer determines not only the customizability of each VN but also the address structure of NVE, the detailed switching mechanism could differ according to the layer. In this Chapter, we consider the network layer virtualization, which is most commonly adopted in the existing network virtualization projects (e.g., Genesis [73], NetScript [74], and X-Bone [75]), and regarded as the most suitable form for supporting the layered Internet architecture.

In order to switch traffic between two different networks virtualized at the network layer, two additional addresses are required: VN ID and VR address. First, VN ID is a unique ID assigned to each VN. The VN ID is used to indicate the neighbor VN in the forwarding table of the switching node. In addition, it can be used as the source address in the tunneling header during traffic switching. Each switched packet can be returned to its original VN by the VN ID written in the source address field of the header. Next, VR address is a fixed node address (such as router ID in OSPF) shared by all virtual nodes hosted on the VR. Since two neighbor VNs do not exchange their routing tables with each other, the switching node has no information about the (neighbor) node address at the returning point. Since VR address is disseminated within every VN by its own routing protocol, the switching node can use this universal address as the indicator of the returning point.

Security is another factor that must be addressed. During the traffic switching process, a plurality of packets are forwarded from one VN to another, so security is of concern to both SPs that operate those VNs. The SP of the switching VN can protect the confidentiality, integrity, and authenticity of the switched packets through a secure tunneling protocol such as IPsec. On the other hand, the SP of the *transit* VN can guarantee the availability of its

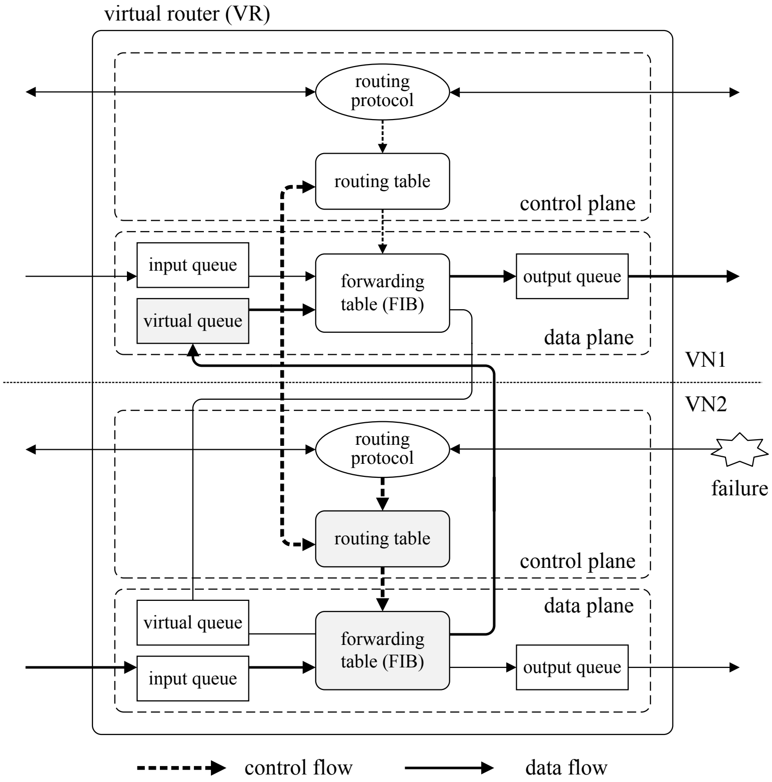


Figure 4.3: Function block diagram of virtual router.

services from an overflow of packets by regulating the peak traffic inflow rate. Moreover, the switched packets can be treated separately for the purpose of security and QoS management since they can be easily recognized with VR addresses.

4.4 Failure Recovery based on Traffic Switching

4.4.1 Inter-VN Traffic Switching

In this section, we present the mechanism of traffic switching between two virtual nodes inside a VR. Figure 4.3 shows the function block diagram of VR.

In this figure, we consider a scenario where a virtual node of VN 2 detects a single link failure and switches the interrupted traffic to VN 1. The functions and flows involved in the inter-VN traffic switching are highlighted with grey boxes and bold arrows. The following events take place in order.

Routing table update

As soon as detecting a failure, the virtual node overwrites the outgoing interfaces with the ID of the neighbor VN for all entries pointing to the failed node or link in the routing table. Here, their metrics are set to as high a value as possible (where lower is better), which aims to immediately replace them with the new detour paths within their VN after the routing re-convergence is completed at all nodes in the VN. Before updating the routing table, the virtual node can optionally query the neighbor virtual node as to whether the returning points of the switched packets are reachable or not over the neighbor VN. If there is no entry for the returning points of some packets in the routing table of the neighbor VN, the virtual node does not update the corresponding entries in its routing table.

Encapsulation

Once the new entries pointing to the neighbor VN are made in the routing table, the forwarding table is updated accordingly. Then, all the following packets that are to be switched to the neighbor VN are wrapped with an additional header which specifies the appropriate returning point for each packet, and forwarded over the neighbor VN. In this process, packet fragmentation will be performed, if necessary, to handle the oversized packets.

Rate control

The encapsulated packets are buffered in the virtual queue of the neighbor virtual node. In the virtual queue, the switched traffic is shaped by a leaky bucket regulator before entering the neighbor VN. The drain rate of the leaky bucket, or the maximum traffic inflow rate into the neighbor VN, is strongly associated with the QoS of the existing traffic in the neighbor VN, therefore being bounded by the SLA and being under control of the neighbor SP.

4.4.2 Failure Recovery Process

We now describe the whole process of the failure recovery based on the inter-VN traffic switching.

Phase 1. Failure detection

Once a node or link failure occurs on a VN, the adjacent nodes are notified of the failure by the absence of *hello* packets of the routing protocol running on the VN. The *hello* packets are sent at a pre-configured interval. If a router does not receive a *hello* packet from a neighbor router within the *dead* interval (the default is four times the value of the *hello* interval), it declares that interface to be down. The failure detection process can be accelerated by reducing the *hello* interval, but too frequent advertisements will introduce routing instability in the network and increase the load on all routers. The fine-tuning of *hello* interval allows failure detection within a few milliseconds. Other alternatives such as bidirectional forwarding detection (BFD) protocol can be used to shorten the detection time.

Phase 2. Traffic switching

The virtual node which has detected the failure immediately begins traffic switching process. Since the inter-VN traffic switching does not require the computation (or selection) of the paths along which the switched packets are delivered to their returning points over the neighbor VN, and the packet encapsulation is performed at wire speed in current high-end routers, with no delay, the forwarding of the interrupted traffic is restored after the failure detection. On the neighbor VN, the switched packets are forwarded along the shortest paths according to the (neighbor) routing table.

Phase 3. Routing re-convergence

Independently of the traffic switching process, the virtual node periodically broadcasts (or multicasts) routing information to other nodes within its VN. All nodes re-compute their routing tables upon receiving the updated routing information. Since the paths via other VN have the highest metric values, they will be overwritten by the new detour paths. The routing re-convergence time is highly dependent upon the number of network prefixes, network topology, and processor speeds.

Phase 4. Detour routing

After the routing re-convergence, all packets are forwarded along their new shortest paths which do not involve the failed node or link, and no more packets are switched to the neighbor VN.

As a result, the inter-VN traffic switching is completed in a short period during routing re-convergence. Thus,

4.5 Numerical Analysis

In this section, we investigate the effect of traffic switching on the QoS of the neighbor VN through the use of analytic performance models based on queuing theory. Although the queuing models do not accurately capture the nature of real Internet traffic, they will provide a useful guide in determining the switching requirements specified in SPA.

We consider two different performance measures: delay and congestion probability. We attempt to follow the most general approaches for these measures. For simplicity of exposition, we confine our attention to the case where the switched traffic travels only a single hop over the neighbor VN.[8] This single-link analysis can be easily extended to the multi-hop switching cases or the analysis of the end-to-end QoS.

4.5.1 Delay

We first analyze the delay performance of the neighbor VN. We model a (virtual) node as an $M/G/1$ queue, where packets arrive according to a Poisson process with rate λ, and the service time has a general distribution (e.g., heavy-tailed distribution) with finite mean $1/\mu$. By the well-known Pollaczek-Khinchine formula, the average queuing delay of each packet is defined as

$$W = \frac{\rho\left(1 + C_v^2(S)\right)}{2\mu\left(1-\rho\right)}, \tag{4.1}$$

where $\rho = \lambda/\mu$, and $C_v(S)$ is the coefficient of variance of the service time distribution S.

Now we consider the situation where some switched traffic is entered to the virtual node. Recall that the switched traffic is shaped by a leaky bucket regulator in the virtual queue. Thus, it does not arrive at the node according to a Poisson process. In fact, the regulated traffic cannot be explicitly modeled

---

[8] This case does not exactly correspond to the next-hop return strategy, which may use more than one hop according to the (neighbor) routing table.

as any random process. Instead, in order to obtain the upper bound of the delay, we assume the worst case that the switched traffic arrives uniformly at the peak switching rate. Let α denote the proportion of the peak switching rate to the average normal arrival rate. Then, the mean and variance of the interarrival time distribution X' during traffic switching is given by

$$E(X') = \frac{1}{\lambda'} = \frac{1}{(1+\alpha)\lambda}, \tag{4.2}$$

$$Var(X') = \frac{1}{\lambda^2}. \tag{4.3}$$

Based on these results, we can generalize our analysis to a $GI/G/1$ queue. The $GI/G/1$ queuing model is difficult to analyze, and up to now, no exact expression of performance measures is known. Fortunately, however, we can use an approximation formula for the average waiting time of the $GI/G/1$ queue presented in [76].[9] Then, the average delay is updated as

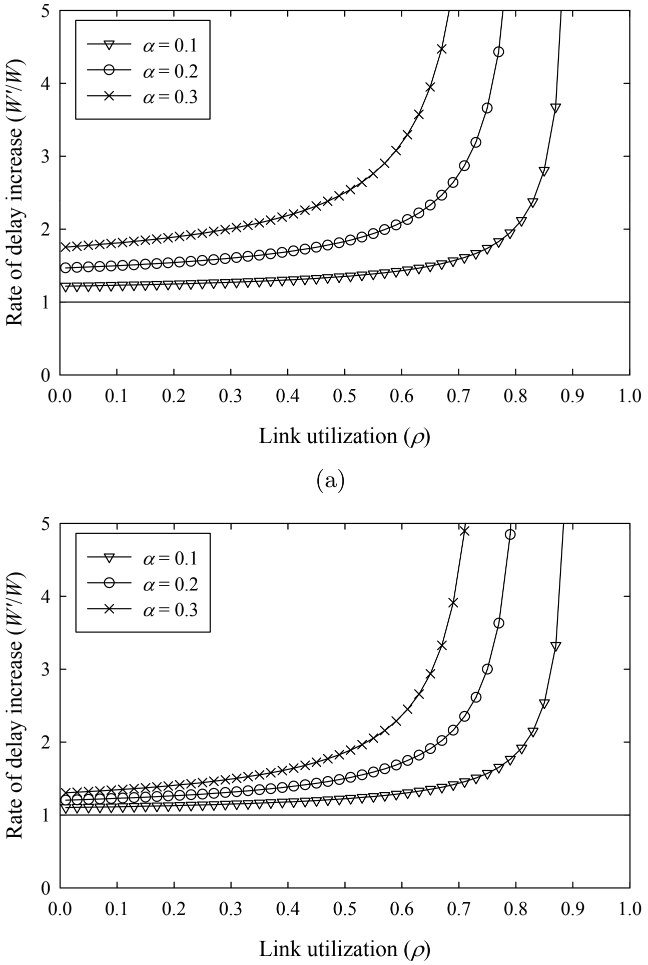
$$\begin{aligned} W' &\approx \frac{\rho'\left(\lambda'^2 Var(X') + \mu^2 Var(S)\right)}{2\mu\left(1-\rho'\right)} \\ &= \frac{(1+\alpha)\,\rho\left((1+\alpha)^2 + C_v^2(S)\right)}{2\mu\left(1-(1+\alpha)\,\rho\right)}, \end{aligned} \tag{4.4}$$

where $\rho' = (1+\alpha)\,\rho$.

In queuing theory, ρ is interpreted as the fraction of time that server is busy. Let q and C denote the average traffic load and link capacity, respectively. Then, it is easy to see that ρ is mapped to the link utilization (i.e., $\rho = q/C$).

---

[9] If the arrivals follow a Poisson process, this approximation formula becomes exactly the same as the Pollaczek-Khinchine formula in (4.1)



(a)

(b)

Figure 4.4: The rate of delay increase for (a) exponential and (b) heavy-tailed service time distributions.

Now, for a given link utilization ρ, we can compute the rate of delay increase as follows:

$$\frac{W'}{W} = \left(1 + \frac{\alpha}{1 - (1+\alpha)\,\rho}\right) \cdot \frac{(1+\alpha)^2 + C_v^2(S)}{1 + C_v^2(S)}. \tag{4.5}$$

Based on this equation, we plot the rate of delay increase for several different values of α in Figure 4.4. Figures 4.4(a) and 4.4(b) show the cases when the service time distributions are exponential (i.e., $C_v^2(S) = 1$) and heavy-tailed (i.e., $C_v^2(S) \gg 1$), respectively. Notice here that in the case of heavy-tailed service time distribution, the second term in the right-hand side of the equation (4.5) approximates to 1. That is, the variance of the service time distribution does not affect the rate of delay increase in the neighbor VN.

As shown in Figure 4.4(b), the rate of delay increase in the case of heavy-tailed service time distribution is very sensitive to a small increase in the average link utilization. Thus, we need to pay careful attention to determining the peak switching rate for each link in order to minimize the QoS degradation due to traffic switching. We can use these results in the following manners: First, the graphs can be used to determine the appropriate peak switching rate for each virtual link depending on the average link utilization of the link when there is a given requirement for the rate of delay increase. Next, they can also be used to estimate the delay increase for given peak switching rate and average link utilization.

4.5.2 Congestion probability

Next, we examine the congestion probability of the *merging* link. The congestion probability is defined as the probability that the traffic load exceeds the target utilization [25]. Since TCP is very sensitive even to congestion at a single link, most ISPs control the average utilization for every link at a certain level. Thus, of practical importance is to analyze the effect of traffic switching on the link congestion probability.

In order to model the congestion probability, we here consider $M/G/\infty$ queuing model. Since the $M/G/\infty$ queue asymptotically captures the selfsimilar property of the Internet traffic, it is widely employed as a realistic traffic model for the Internet in many studies [27,29]. In the $M/G/\infty$ traffic model, the network itself is regarded as a queuing system with an infinite number of servers. At the end points, packets are generated according to a Poisson process with mean λ, and the service time for each packet has a heavy-tailed distribution with mean $1/\mu$. Then, the number of packets in the system (i.e., the network) at a certain time has a Poisson marginal distribution with mean λ/μ [30]. Let q_r denote the average traffic load of an end-to-end route r. Since traffic loads along all the routes in the network can be assumed to be independent, the number of aggregated packets on a link l at a certain time also follows a Poisson marginal distribution with mean $q_l = \sum_{r:l\in r} q_r$.

Then, for a given average traffic load q and link capacity C of the merging link, the congestion probability in the normal case is given by

$$K = \sum_{k=\eta C+1}^{\infty} \frac{q^k}{k!} e^{-q}, \tag{4.6}$$

where η is the link utilization threshold.

Here we also consider the worst case scenario that the switched traffic is entered at a uniform rate. Since the switched traffic occupies a fixed portion of link capacity with no variation, we can intuitively compute the updated congestion probability during traffic switching by deducting the corresponding portion of capacity from the whole link capacity as follows:

$$K' = \sum_{k=\eta C-\alpha q+1}^{\infty} \frac{q^k}{k!} e^{-q} \tag{4.7}$$

Figure 4.5 shows the congestion probability for several different values of α. The solid line represents the congestion probability in the normal case

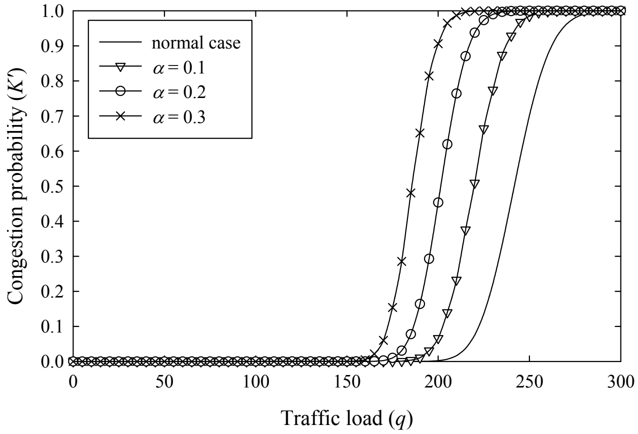


Figure 4.5: Congestion probability.

(i.e., $\alpha = 0$). In this example, the link capacity C is set to 300 (measured in unit of bandwidth per unit time)[10], and the link utilization threshold η is set to 0.8. Like the rate of delay increase, the congestion probability is highly sensitive to the change of average traffic load. We can apply this result to find the optimal peak switching rate or to estimate the increase in congestion probability during the traffic switching phase.

4.6 Summary

In this Chapter, we developed a fast failure recovery mechanism which cleverly exploits the structural feature of NVE. Despite its considerable advantages compared to the existing IP fast reroute, the failure recovery through inter-VN traffic switching has a potential risk that the switched traffic could cause QoS degradation in the neighbor VN. Thus, we analyzed the possible adverse effect of traffic switching on both the rate of delay increase and the congestion

---

[10] Since the congestion probability is granularity-sensitive, it is important to set the size of link capacity in unit of bandwidth.

probability based on queuing theoretic models, and provided useful guidelines to determine the optimal peak switching rate within the acceptable range of QoS degradation.

We believe that the theoretical study in this Chapter is just the first step to develop a practical failure recovery protocol. We need to further define more detailed requirements of the inter-VN traffic switching process, and verify the efficacy of the proposed failure recovery mechanism in more various ways. Also, we should evaluate the real performance of the proposed mechanism (e.g., end-to-end delay, packet loss, and forwarding discontinuity) by conducting experiments on a VN testbed. We are currently working on running the proposed failure recovery mechanism on GENI testbed.

Chapter 5

Conclusion

In this dissertation, I focused on several important issues in the network virtualization environment. I tried to raise new issues and take different approaches to the problems. In Chapter 2, I proposed an efficient and effective resource allocation method to offer differentiated service in virtual networks. In Chapter 3, I developed an economic model to analyze the user subscription behavior and the ISP revenue of tiered access service in virtual networks. In Chapter 4, I designed a fast failure recovery mechanism through inter-virtual network traffic switching in virtual networks. These studies have a great contribution in terms of addressing several practical problems to build a virtual network-based Internet architecture. Although some parts of the studies remain unsolved and need further attention in the future, I believe that the theoretical analysis and outstanding results in this dissertation will definitely provide useful guidelines to improve the current Internet and to design the Future Internet architecture.

Appendix A

Proofs of Lemmas

A.1 Proof of Lemma 2

We prove this lemma by using a simple graphical presentation. From equation (3.13), the steady-state number of subscribers n^* that $n^* > C/r$ can be obtained at the point of contact between two lines $y = n$ and $y = N\left(1 - F_v\left(pn/C\right)\right)$ as described in Figure A.1.

First, for the given price p_1, let n_1^* denote the corresponding steady-state number of subscribers. Then

$$n_1^* = N\left(1 - F_v\left(p_1 n_1^*/C\right)\right). \tag{A.1}$$

Let us now consider the case for the price p_2 that satisfies $p_2 = \alpha p_1$ and $\alpha > 1$. As depicted in the figure, the decreasing line drops more rapidly than that for p_1. Let n_2' denote the value of n at the point of contact with the horizontal extension line of n_1^*. Here, n_2' satisfies $n_2' = n_1^*/\alpha$.

Then the steady-state number of subscribers n_2^* which is given by the point of contact between $y = n$ and $y = N\left(1 - F_v\left(p_2 n/C\right)\right)$ always satisfies

$$n_2^* > n_1^*/\alpha \tag{A.2}$$

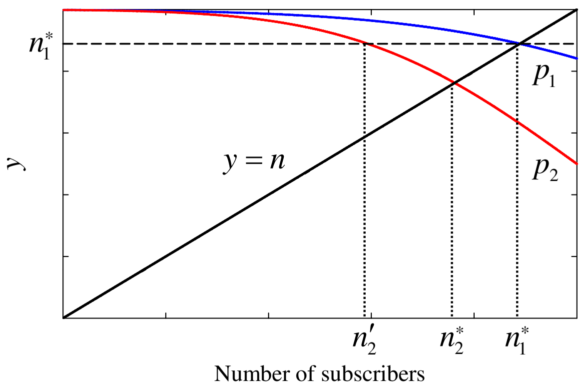


Figure A.1: Graphical presentation of finding the steady-state number of subscribers.

due to the fact that the latter line is a monotone decreasing function of n.

Thus, the ISP revenue given by $R = n^* p$ satisfies the following condition for two prices p_1 and p_2:

$$R_2 = n_2^* p_2 > n_1^* p_1 = R_1 \tag{A.3}$$

That is, the ISP revenue is an increasing function of p regardless of $F_v(\cdot)$.

□

A.2 Proof of Lemma 3

The proof of this lemma is straightforward. We can verify that the maximum ISP revenue occurs in the *regulated range*, if the ISP revenue at $p = p_h$ is always larger than that of p_l, which is the maximum value in the *decreasing range*. First, we can easily obtain the ISP revenue at p_h. Since the steady-state number of subscribers at this price is C/r, the ISP revenue is given by

$$R\left(p_h\right) = C F_v^{-1}\left(1 - \frac{C}{Nr}\right). \tag{A.4}$$

The ISP revenue at p_l, however, is not as clearly expressed in a closed form as $R(p_h)$. Instead, we can define the boundary condition for $R(p_l)$. Recall the mechanism of the oscillation in the number of subscribers described in Figure 3.3. If the access price is set to the lower-bound value p_l, the number of subscribers oscillates continuously between C/r and $\tilde{n}$. Let p_l^- denote the access price which is very close to but slightly smaller than p_l. Since p_l^- does not belong to the oscillation range, the number of subscribers converges to a unique value $n^*(p_l^-)$, which definitely satisfies $C/r < n^*(p_l^-) < \tilde{n}$. The ISP revenue at p_l therefore always satisfies the following inequality:

$$R\left(p_l\right) \simeq n^*(p_l^-)p_l = \frac{n^*(p_l^-)}{\tilde{n}} C F_v^{-1}\left(1 - \frac{C}{Nr}\right) < R\left(p_h\right)$$

From this, we complete the proof. □

Bibliography

[1] B. Chun, D. Culler, T. Roscoe, A. Bavier, L. Peterson, M. Wawrzoniak, and M. Bowman, "PlanetLab: An Overlay Testbed for Broad-Coverage Services," ACM SIGCOMM Computer Communication Review, vol. 33, no. 3, pp. 3–12, 2003.

[2] GENI: Global Environment for Network Innovations. http://www.geni.net/.

[3] T. Anderson, L. Peterson, S. Shenker, and J. Turner, "Overcoming the Internet Impasse through Virtualization," IEEE Computer, vol. 38, no. 4, pp. 34–41, 2005.

[4] J. Turner and D. Taylor, "Diversifying the Internet," in *Proc. IEEE GLOBECOM*, 2005.

[5] N. Feamster, L. Gao, and J. Rexford, "How to Lease the Internet in Your Spare Time," ACM SIGCOMM Computer Communication Review, vol. 37, no. 1, pp. 61–64, 2007.

[6] J. Fan and M. Ammar, "Dynamic Topology Configuration in Service Overlay Networks - a Study of Reconfiguration Policies," in *Proc. IEEE INFOCOM*, 2006.

[7] Y. Zhu and M. Ammar, "Algorithms for Assigning Substrate Network Resources to Virtual Network Components," in *Proc. IEEE INFOCOM*, 2006.

[8] M. Yu, Y. Yi, J. Rexford, and M. Chiang, "Rethinking Virtual Network Embedding: Substrate Support for Path Splitting and Migration," ACM SIGCOMM Computer Communication Review, vol. 38, no. 2, pp. 17–29, 2008.

[9] N. Chowdhury, M. Rahman, and R. Boutaba, "Virtual Network Embedding with Coordinated Node and Link Mapping," in *Proc. IEEE INFOCOM*, 2009.

[10] I. Houidi, W. Louati, W. Ameur, and D. Zeghlache, "Virtual Network Provisioning across Multiple Substrate Networks," Elsevier Computer Networks, vol. 55, no. 4, pp. 1011–1023, 2011.

[11] X. Cheng, S. Su, Z. Zhang, H. Wang, F. Yang, Y. Luo, and J. Wang, "Virtual Network Embedding through Topology-Aware Node Ranking," ACM SIGCOMM Computer Communication Review, vol. 41, no. 2, pp. 39–47, 2011.

[12] A. Ludwig, S. Schmid, and A. Feldmann, "The Price of Specificity in the Age of Network Virtualization," in *Proc. IEEE/ACM UCC*, 2012.

[13] G. Schaffrath, S. Schmid, and A. Feldmann, "Optimizing Long-Lived CloudNets with Migrations," in *Proc. IEEE/ACM UCC*, 2012.

[14] N. M. M. K. Chowdhury and R. Boutaba, "Network Virtualization: State of the Art and Research Challenges," IEEE Communications Magazine, vol. 47, no. 7, pp. 20–26, 2009.

[15] R. Braden, D. Clark, and S. Shenker, "Integrate Services in the Internet Architecture: Overview," RFC 1633, 1994.

[16] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin, "Resource ReSerVation Protocol (RSVP)," RFC 2205, 1997.

[17] S. Shenker, C. Partridge, and R. Guerin, "Specification of Guaranteed Quality of Service," RFC 2212, 1997.

[18] F. Baker, C. Iturralde, F. Le Faucheur, and B. Davie, "Agregation of RSVP for IPv4 and IPv6 Reservations," RFC 3175, 2001.

[19] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An Architecture for Differentiated Services," RFC 2475, 1998.

[20] D. Clark, J. Wroclawski, K. Sollins, and R. Braden, "Tussle in Cyberspace: Defining Tomorrows Internet," in *Proc. ACM SIGCOMM*, 2002.

[21] A. Bavier, N. Feamster, M. Huang, L. Peterson, and J. Rexford, “In VINI Veritas: Realistic and Controlled Network Experimentation,” in *Proc. ACM SIGCOMM*, 2006.

[22] W. Szeto, Y. Iraqi, and R. Boutaba, “A Multi-Commodity Flow Based Approach to Virtual Network Resource Allocation,” in *Proc. IEEE GLOBECOM*, 2003.

[23] J. Lu and J. Turner, “Efficient Mapping of Virtual Networks Onto a Shared Substrate,” Washington University, Tech. Rep. WUCSE-2006-35, 2006.

[24] J. Lischka and H. Karl, “A Virtual Network Mapping Algorithm based on Subgraph Isomorphism Detection,” in *Proc. ACM VISA*, 2009.

[25] Z. Duan, Z. Zhang, and Y. Hou, “Service Overlay Networks: SLAs, QoS, and Bandwidth Provisioning,” IEEE/ACM Transations on Networking, vol. 11, no. 6, pp. 870–883, 2003.

[26] W. Leland, M. Taqqu, W. Willinger, and D. Wilson, “On the Selfsimilar Nature of Ethernet Traffic,” IEEE/ACM Transactions on Networking, vol. 2, no. 1, pp. 1–15, 1994.

[27] V. Paxon and S. Floyd, “Wide Area Traffic: The Failure of Poisson Modeling,” in *Proc. ACM SIGCOMM*, 1994.

[28] M. Crovella and A. Bestavros, “Self-Similarity in World Wide Web Traffic: Evidence and Causes,” in *Proc. ACM SIGMETRICS*, 1996.

[29] M. Parulekar and A. Markowski, “M/G/∞ Input Processes: A Versatile Class of Models for Network Traffic,” in *Proc. IEEE INFOCOM*, 1997.

[30] D. Cox and V. Isham, Point Processes. London, U.K.: Chapman Hall, 1980.

[31] F. Kelly, “Routing in Circuit-Switched Networks: Optimization, Shadow Prices and Decentralization,” Advances in Applied Probability, vol. 20, no. 1, pp. 112–144, 1988.

[32] A. Schrijver, Theory of Linear and Integer Programming. NewYork, NY, USA: John Wiley & Sons, Inc., 1986.

[33] R. Ahuja, T. Magnanti, and J. Orlin, Network Flows: Theory, Algorithms, and Applications. Englewood Cliffs, NJ, USA: Prentice Hall, 1993.

[34] C. Barnhart, C. Hane, and P. Vance, "Using Branch-and-Price-and-Cut to Solve Origin-Destination Integer Multicommodity Flow Problems," Operationss Research, vol. 48, no. 2, pp. 318–326, 2000.

[35] COIN-OR Project https://projects.coin-or.org/Bcp.

[36] R. B. Wilson, Nonlinear Pricing. Oxford University Press, 1997.

[37] K. T. Talluri and G. van Ryzin, The Theory and Practice of Revenue Management. Springer, 2005.

[38] L. He and J. C. Walrand, "Pricing Differentiated Internet Services," in *Proc. IEEE INFOCOM*, 2005.

[39] G. Kesidis, A. Das, and G. de Veciana, "On Flat-rate and Usage-based Pricing for Tiered Commodity Internet Services," in *Proc. IEEE CISS*, 2008.

[40] Q. Lv and G. N. Rouskas, "An Economic Model for Pricing Tiered Network Services," in *Proc. IEEE ICC*, 2009.

[41] G. N. Rouskas and N. Baradwaj, "On Bandwidth Tiered Service," IEEE/ACM Transactions on Networking, vol. 17, no. 6, pp. 1780–1793, 2009.

[42] A. Odlyzko, "Paris Metro Pricing for the Internet," in *Proc. ACM Conference on Electronic Commerce*, 1999.

[43] M. Falkner, M. Devetsikiotis, and I. Lambadaris, "An Overview of Pricing Concepts for Broadband IP Networks," IEEE Communications Surveys, vol. 3, no. 2, pp. 2–13, 2000.

[44] D. Allan, J. Kempf, and T. Cagenius, "Enabling the Network-Embedded Cloud," Ericsson Review, no. 2, pp. 35–39, 2012.

[45] C.-K. Chau, Q. Wang, and D.-M. Chiu, "On the Viability of Paris Metro Pricing for Communication and Service Networks," in *Proc. IEEE INFOCOM*, 2010.

[46] S. Ren, J. Park, and M. van der Schaar, “User Subscription Dynamics and Revenue Maximization in Communications Markets,” in *Proc. IEEE INFOCOM*, 2011.

[47] D. Besanko, S. Gupta, and D. Jain, “Logit Demand Estimation Under Competitive Pricing Behavior: An Equilibrium Framework,” Management Science, vol. 44, no. 11, pp. 1533–1547, 1998.

[48] V. Valancius, C. Lumezanu, N. Feamster, R. Johari, and V. Vazirani, “How Many Tiers? Pricing in the Internet Transit Market,” in *Proc. ACM SIGCOMM*, 2011.

[49] R. Jain, T. Mullen, and R. Hausman, “Analysis of Paris Metro Pricing Strategy for QoS with a Single Service Provider,” in *Proc. IWQoS*, 2001.

[50] David Ros and Bruno Tuffin, “A Mathematical Model of the Paris Metro Pricing Scheme for Charging Packet Networks,” Computer Networks, vol. 46, no. 1, pp. 73–85, 2004.

[51] R. Gibbens, R. Mason, and R. Steinberg, “Internet Service Classes Under Competition,” IEEE Journal on Selected Areas in Communications, vol. 18, no. 12, pp. 2490–2498, 2000.

[52] Y. Jin, S. Sen, R. Guérin, K. Hosanagar, and Z.-L. Zhang, “Dynamics of Competition Between Incumbent and Emerging Network Technologies,” in *Proc. NetEcon*, 2008.

[53] S. Boyd and L. Vandenberghe, Convex Optimization. Cambridge University Press, 2004.

[54] P. Laskowski and J. Chuang, “Innovations and Upgrades in Virtualized Network Architectures,” in *Proc. NetEcon*, 2010.

[55] A. Atlas and A. Zinin, “Basic Specification for IP Fast Reroute: Loop-Free Alternates,” RFC 5286, Sep. 2008.

[56] A. Markopoulou, G. Iannaccone, S. Bhattacharyya, C. Chuah, Y. Ganjali, and C. Diot, “Characterization of Failures in an Operational IP Backbone Network,” IEEE/ACM Transactions on Networking, vol. 16, no. 4, pp. 749–762, 2008.

[57] A. Raj and O. C. Ibe, “A Survey of IP and Multiprotocol Label Switching Fast Reroute Schemes,” Elsevier Computer Networks, vol. 51, no. 8, pp. 1882–1907, 2007.

[58] P. Francois, C. Filsfils, J. Evans, and O. Bonaventure, “Achieving Sub-Second IGP Convergence in Large IP Networks,” ACM SIGCOMM Computer Communication Review, vol. 35, no. 2, pp. 35–44, 2005.

[59] A. Basu and J. G. Riecke, “Stability Issues in OSPF Routing,” in *Proc. ACM SIGCOMM*, 2001.

[60] M. Shand and S. Bryant, “IP Fast Reroute Framework,” RFC 5714, Jan. 2010.

[61] S. Bryant, M. Shand, and S. Previdi, “IP Fast Reroute Using Not-via Addresses,” Internet Draft, draft-ietf-rtgwg-ipfrr-notvia-addresses-09, Jun. 2012.

[62] S. Nelakuditi, S. Lee, Y. Yu, Z.-L. Zhang, and C.-N. Chuah, “Fast Local Rerouting for Handling Transient Link Failures,” IEEE/ACM Transactions on Networking, vol. 15, no. 2, pp. 359–372, 2007.

[63] Z. Zhong, S. Nelakuditi, Y. Yu, S. Lee, J. Wang, and C.-N. Chuah, “Failure Inferencing based Fast Rerouting for Handling Transient Link and Node Failures,” in *Proc. IEEE INFOCOM*, 2005.

[64] A. Kvalbein, A. F. Hansen, T. Ĉiĉić, S. Gjessing, and O. Lysne, “Multiple Routing Configurations for Fast IP Network Recovery,” IEEE/ACM Transactions on Networking, vol. 17, no. 2, pp. 473–486, 2009.

[65] T. Ĉiĉić, A. F. Hansen, and O. K. Apeland, “Redundant Trees for Fast IP Recovery,” in *Proc. BROADNETS*, 2007.

[66] P. Pan, G. Swallow, and A. Atlas, “Fast Reroute Extensions to RSVP-TE for LSP Tunnels,” RFC 4090, May 2005.

[67] K.-W. Kwong, L. Gao, R. Guerin, and Z.-L. Zhang, “On the Feasibility and Efficacy of Protection Routing in IP Networks,” in *Proc. IEEE INFOCOM*, 2010.

[68] G. Rétvári, J. Tapolcai, G. Enyedi, and A. Császár, “IP Fast ReRoute: Loop Free Alternates Revisited,” in *Proc. IEEE INFOCOM*, 2011.

[69] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris, “Resilient Overlay Networks,” in *Proc. ACM Symposium on Operating Systems Principles*, 2001.

[70] J. He, R. Zhang-Shen, Y. Li, C.-Y. Lee, J. Rexford, and M. Chiang, “DaVinci: Dynamically Adaptive Virtual Networks for a Customized Internet,” in *Proc. ACM CoNEXT*, 2008.

[71] UCLP: User Controlled Lightpaths. http://www.uclp.ca/.

[72] P. Ruth, X. Jiang, D. Xu, and S. Goasguen, “Virtual Distributed Environments in a Shared Infrastructure,” Computer, vol. 38, no. 5, pp. 63–69, 2005.

[73] M. E. Kounavis, A. T. Campbell, S. Chou, F. Modoux, J. Vicente, and H. Zhuang, “The Genesis Kernel: A Programming System for Spawning Network Architectures,” IEEE Journal on Selected Areas in Communications, vol. 19, no. 3, pp. 511–526, 2001.

[74] S. da Silva, Y. Yemini, and D. Florissi, “The NetScript Active Network System,” IEEE Journal on Selected Areas in Communications, vol. 19, no. 3, pp. 538–551, 2001.

[75] J. Touch, “Dynamic Internet Overlay Deployment and Management Using the X-Bone,” Computer Networks, vol. 36, no. 2/3, pp. 117–135, 2001.

[76] J. F. C. Kingman, “A Martingale Inequality in the Theory of Queues,” Mathematical Proceedings of the Cambridge Philosophical Society, vol. 60, no. 2, pp. 359–361, 1964.

국문초록

네트워크 가상화는 물리적 네트워크의 공유 자원들을 복수 개의 가상 네트워크들에 동적으로 할당할 수 있게 해주는 기술이다. 자원 할당의 유연성과 가상 네트워크들 사이의 독립성 때문에, 네트워크 가상화는 네트워크 테스트베드를 설계하기 위한 기반 기술로써 주로 활용되어 왔을 뿐만 아니라, 인터넷의 다양화를 지원하기 위한 비용 효율 높은 해결책으로써 여겨지기 시작했다. 서비스에 따라 계층화된 인터넷을 설계하기 위한 하나의 수단으로써, 네트워크 가상화는 여전히 해결해야 할 많은 도전 과제들을 가지고 있다. 이 학위 논문은 가상 네트워크 환경에서 중요한 몇 가지 새로운 연구 주제들을 제시하고, 그에 대한 효과적인 해법들을 제안한다.

첫 번째로, 가상 네트워크의 다양한 QoS 요구사항을 만족시킬 수 있는 네트워크 최적 분할 방법을 제안한다. QoS와 대역폭 제한 조건을 고려하여 가상 네트워크 분할 문제를 최적화 문제로 모형화하고, 문제의 구조적 복잡성을 해결하기 위해 최단 경로 라우팅에 기반한 휴리스틱을 제안한다. 실제 인터넷 환경을 고려한 대규모 실험을 통해, 제안한 휴리스틱의 효율성과 확장성을 입증한다.

다음으로, 가상 네트워크에서 차등 접속 서비스를 위한 경제성 분석 모델을 제시한다. 먼저 사용자 가입 변동 모형이 한 값으로 수렴하기 위한 충분 조건을 유도하고, 이러한 조건 하에서 인터넷 서비스 제공자의 수익을 최대화할 수 있는 최적의 가격 결정 방법 및 대역폭 분할 방법을 찾는다. 수치 실험을 통해, 적절한 가격 결정과 대역폭 분할이 이루어진다는 가정 하에서 차등화 서비스가 단일 서비스보다 더 높은 수익성을 나타낼 수 있음을 증명한다.

마지막으로, 가상 네트워크 간 트래픽 전환을 통한 빠르고 효과적인 고장 회복 기술을 개발한다. 가상 네트워크의 구조적 특성을 활용한 고장 회복 기술을 이용하면, 모든 링크에 대한 백업 경로가 항상 존재하도록 미리 토폴로지를 설계해야 할 필요가 없고, 각 라우터에서 그 경로들에 대한 계산을 미리 해 놓을 필요가 없다. 그럼에도 불구하고, 제안한 고장 회복 방법은 기존의 기술들과 같은 좋은 성능을 보인다.

이 학위 논문은 가상 네트워크를 기반으로 하는 인터넷 환경에서 발생할 수 있는 중요한 문제들을 다루고자 한다. 이 논문에서 제안하는 분석 모델 및 실험 결과들은 현재 인터넷의 한계를 극복하고, 미래 인터넷 아키텍처를 설계하기 위한 유용한 지침을 제공할 것이다.

주요어: 네트워크 가상화, 서비스 차등화, 차등 접속 서비스, 가격 결정, 대역폭 분할, 고장 회복

학 번: 2007-21044