University of Nevada, Reno

Link State Contract Routing

A thesis submitted in partial fulfillment of the
requirements for the degree of Master of Science in
Computer Science

by

Hasan T. Karaoglu

Dr. Murat Yuksel/Thesis Advisor

Dec, 2009

THE GRADUATE SCHOOL

We recommend that the thesis
prepared under our supervision by

HASAN T. KARAOGLU

entitled

Link State Contract Routing

be accepted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE

Murat Yuksel, Ph. D., Advisor

Mehmet H. Gunes, Ph. D., Committee Member

Tigran Melkonyan, Ph. D., Graduate School Representative

Marsha H. Read, Ph. D., Associate Dean, Graduate School

December, 2009

Link State Contract Routing Protocol

Hasan T. Karaoglu

University of Nevada, Reno, 2009

Supervisor: Murat Yuksel

Abstract

The Internet's simple design resulted in huge success in basic telecommunication services. However, the current Internet architecture has failed in terms of introducing many innovative technologies as end-to-end (E2E) services such as multicasting, guaranteed quality of services (QoS) and many others. We argue that contracting over static service level agreements (SLA) and point-to-anywhere service definitions are the main reasons behind this failure. In that sense, the Internet architecture needs major shifts since it neither allows (i) users to indicate their value choices at sufficient granularity nor (ii) providers to manage risks involved in investment for new innovative QoS technologies and business relationships with other providers as well as users. To allow these much needed economic flexibilities, we introduce contract-switching as a new paradigm for the design of future Internet architecture. In this work, we implement contract-routing framework with specific focus on long-term contracted services in Link State Contract Routing scheme. Our work shows that E2e guaranteed QoS services can be achieved in routing over contracted edge-to-edge service abstractions which are built on today's popular protocols with reasonable protocol overhead.

Contents

List of Figures

Chapter 1

Introduction

1.1 Internet Architecture and Current Issues

The Internet's simplistic design which seeks simplicity at frequently used core entities and increasing complexity (when necessary) at specialized edge entities definitely [34] constitutes one of the factors behind the huge success of the Internet so far. By following this simple yet powerful "End-to-End Argument", designers have targeted to keep the Internet core *transparent* and *simple* as if its only task would be letting packets in and out [10, 18]. The intuition behind these targets is that if core of the Internet is kept simple and less bounded to application-specific functionalities, introducing new applications and protocols (or upgrading and replacing old ones as well) will be much more easier. So, the Internet will always be open to innovative killer applications, protocols and stay evolvable [10, 18].

Following these principles, Internet has been really successful to deliver its promises in terms of basic communication services and evangelizing communication technologies in people's daily life irreversibly. Despite its success of its success of

delivering basic communication services, the current Internet architecture has failed to introduce guaranteed QoS as an end-to-end service [10]. Another shortcoming of today's Internet architecture is lacking of flexible business settlements which compensate providers for their risk-taking in delivering innovative services (i.e. QoS, multicast) beyond basic reachability [23]. In a commercial market like today's Internet, providers have to make profit out of their investment. Before making investment on new technologies, they need more incentives to risk their money on providing services on risky enhanced technologies rather than playing safe and paying for equipments of less risky and strictly controlled best-effort basic communication services. We claim that today's Internet architecture failed to incentivize providers to invest on innovative communication services [10, 23, 48]. In this sense, we can claim that lacking of flexible business models hinder evolvability target of the "end-to-end argument".

In the absence of flexible business models which allow service providers to hedge their risks, they want to control their networks and services strictly in a sense to reduce unpredictability. This policy of providers leaves only little chance for customers to express their value choices. Today, an enterprise company that needs bandwidth guarantees to an arbitrary point in the Internet for a short period of time, does not have a way to express its needs [23]. Similarly, a home user living in the United States wants to guarantee its video quality while watching a soccer game in Turkey can't close the deal with its provider for a temporary service upgrade specifically for the duration of the game.

Although the examples above look trivial, lack of user choice expression capability tools have effects beyond a disappointment of a soccer team fan or frustrated company employees due to low quality Virtual Private Network (VPN) connection to their remote office in Japan. Considering commercialization of the Internet, every

action on the virtual world like clicking a website has tangible effects in the real world. Either in terms of advertising revenues or traffic flows in and out of provider network boundaries result in value flows which mean money flows in and out pockets of content providers, Internet service providers (ISP), advertisement companies or home users. Stakeholders' efforts of managing these value flows on the Internet has both success stories as can be seen on emergence of companies like Google, Yahoo, Akamai and also failure stories as can be seen on demise of dot-com companies and ISPs after dot-com boom. Today, stakeholders of the Internet have different interests on how to manage these value flows for their own benefits and these have lead emergence of tussles which shape the Internet [10]. In the absence of value choice capability for customers, customers are not able to express their choices in these tussles effectively. User choice definitely drives innovation and product enhancement, and imposes discipline on the marketplace [10]. Without user choice, providers also lose their ability to capture changes and trends in user demand on their services which reduce their capability to manage risks.

Although handicapped by limited expression power, user choice and specifically customer demand for guaranteed QoS services can be observed from the existence of voice-over-IP (VOIP) companies which apply adaptive techniques to emulate end-to-end service quality at the level of voice and video on top of "best-effort" connections. Another clue can be given as the existence of content delivery networks. Content delivery networks simply comprised of hundreds of thousands of geographically scattered routers which cache popular content like videos, movies, and web sites so as to offer load balancing, better reliability, enhanced availability, reduced latency and more quality enhancements to content owners while distributing their contents. Akamai, the leader company in content delivery business today, carries 30 percent of the

Internet traffic all by itself [47]. Finally there are several companies which broker end-to-end connectivity services with guaranteed bandwidth promises by mediating between service providers and enterprise companies e.g., Equinix [15], Arbinet [1].

Telecommunication companies have a requirement inherent to their sector which is to renovate their networks and infrastructure to even maintain their business at current levels. This requirement makes these companies to have a dilemma on how to manage their investments in terms of where to invest, when to invest, and in what scale [47]. Without earnest capabilities to capture user demand, current Internet architecture results in lower social welfare [16,21]. This can be argued because the tussle between providers and customers can be described as a win-win type tussle, which both ends can benefit. Users should pay more to be able to express their value choices where providers have to meet customer demand in return of compensation [10]. To express their value choices, customers need open interfaces offered by providers. For providers to offer these capabilities to their own customers in an end-to-end manner, they have to collaborate and offer these interfaces to other providers first [5]. Previous research shows that, even at small scale, such a collaboration between providers over open interfaces will result in huge gains in quality and economics of Internet routing [13, 25, 37]. Today's Internet benefits from driving and self-correcting force of economics in limited sense. Introducing user value choice and enhanced provider risk management tools via flexible business models will result in diversity in services and products, efficiency, transparency and evolvability which are hindered by the current Internet architecture.

1.2 Motivation and Challenges

Today's de facto inter-domain protocol Border Gateway Protocol (BGP) is the glue of the Internet that holds the Internet together. Designed in compliance with "end-to-end argument", BGP is responsible for the exchange of basic reachability information among autonomous networks which comprise the Internet (a.k.a network of networks). By advertising and filtering reachability information about their neighbors and customers according to their policies, ISP providers try to manage inbound and outbound value flows flowing in and out of their boundaries according to their preferences. According to gathered neighboring relationship map and their business targets, ISPs choose the shortest paths calculated by BGP to their destinations to route their outbound traffic.

BGP does not take neither quality of route nor economic feasibility into account while calculating the shortest path. Since inter-domain routing protocol does not offer any information exchange on quality, price or additional capabilities of routes, service provider companies usually do not offer any differentiated services to particular destinations. Rather indifferent to destinations, they offer point-to-anywhere services whose accounting based on bulk size of exchanged traffic. So, current agreements (a.k.a SLAs) between service providers on the Internet do not support quality of service (QoS) grades beyond basic settings of availability, loss percentage and these are only limited to reach of next hop boundaries (immediate neighbors), not beyond that [23, 48].

Although point-to-anywhere (P2A) service mechanism brings simplicity and ease for accounting and management of traffic flows, by adopting it players lose their chances of exploring better end-to-end paths (both economically and technically).

Analysis made by Teixeira et al. reveals that at intra-domain level nearly 90 percent of entry-exit point pairs of Tier-1 ISPs have multiple disjoint paths [43]. Moreover, at inter-domain level 75 percent of the time, there can be found disjoint end-to-end paths. According to these findings, the Internet offers diversity on both intra-domain and inter-domain level. However, as Savage et al. show that large number of paths on the Internet (up to 80%) have alternate paths that offer better quality as measured by delay, loss and bandwidth [35]. Unfortunately, the current Internet architecture does not allow exploitation of these alternative paths due to provider policies and limitation of shortest path routing.

Besides these issues, since service level agreements (SLA) between stakeholders boil down to simple bandwidth trade contracts lacking dimensions like QoS or economics to manage risks, business models which are built on top of them do not provide enough incentives for ISPs to renovate their infrastructures. In the absence of E2E innovative services offered by infrastructure owners of the core Internet, demand for these services are met by third party companies which offer alternatives to emulate these services [10]. While third party companies make more money over enhanced services, infrastructure owners of Internet core do not get their fair share out of these extra-revenues proportional to their contribution of operating infrastructure carrying them. So, the extra-gains generated by third parties do not provide enough compensation for providers to renovate their infrastructure. Lack of business models and incentive mechanisms compose a big threat on future of the Internet due to this handicap in provider compensation model [3, 14, 47].

As an another issue, time-scales of SLAs are too long (e.g., months to years) and there is typically no way of bailing out of an SLA if the ISP finds a better deal. Further, SLAs are closed at the present time (or very near future such as days/weeks)

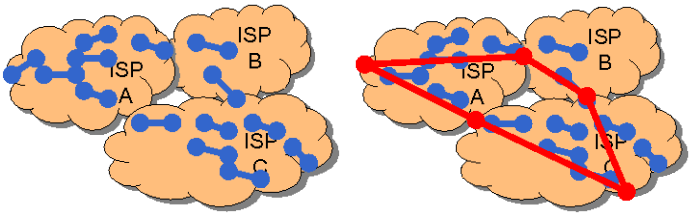
and an ISP typically can not easily close deals for its future investments to reduce risks involved in its investment. It is a pressing need to have such economic instruments for enabling the ISPs to manage risks in their investments [23].

As David Clark proposed, we don't and can't design the answer to solve all inter-domain routing issues or more generally all tussles. Since Internet is shaped by huge number of stakeholders who have different set of interests and motivations, there will be tussle and conflict among these entities regardless of what future Internet design and protocols would be. So, we can't design the answer but "All we can design is space for the tussle." [10].

So, our target is not finding the solution but designing mechanisms that provide enough space and dimensions as commonground for negotiating parties. We are aware of the fact that we trade flexibility in return of increasing complexity. Since we make our design for choice and diversity in outcomes, we increase complexity. However, we believe increasing market efficiency in return of flexible business models will generate enough playground for making these compromises. Our expectation is that economic stabilization mechanism will create its own popular options in service delivery and leave us a feasible set of service options diverse enough to meet market demand but also classified and focused enough to scale and stabilize.

Another tradeoff we have is that our design requires open interfaces for parties to express their choices. More transparency is not pleasant for the service providers traditionally. But, it is necessary also for enabling service providers to advertise cost of diverse set of services in our proposed inter-domain routing architecture.. Yet, we leave enough space for providers to define their offers within the bounds of their required level of confidentiality.

In that sense, we believe that our design draws the line between transparency



(a) Packet-switching (b) Contract-switching

Figure 1.1: Packet Switching and Contract Switching

for confidentiality and complexity for scalability in a realistic manner.

1.3 Contract Routing

We propose an Internet architecture that allows flexible, finer grained, dynamic contracting over multiple providers. With such capabilities, the Internet itself will be viewed as a *"contract-switched"* network beyond its current status as a "packet-switched" network. This contract-switching architecture depends on definition of *"contract links"*. A Contract Link technically represents a service abstraction between edge routers of a domain. This edge-to-edge (G2G) service abstraction is not comprised of a mere domain-level tunneling definition between borders of an ISP but also technical, financial and time components which set the terms for Service Level Agreement (SLA) attached to this virtual tunneling service. Once service providers advertise their capabilities in such contract link advertisements, they become able to compose end-to-end *"contract paths"* by concatenating the contract links advertised by other service providers too. The capability of composing end-to-end paths will let the emergence of a Contract-Switched Architecture, where routing is made according to contracts and established end-to-end (E2E) contract paths rather than individual routing decisions made on routers hop by hop.

We view "contract-switching" as a generalization of the packet-switching paradigm of the current Internet architecture. For example, *size of a packet* can be considered as a special case of the *capacity of a contract* to expire at a very short-term, e.g. transmission time of a packet. Similarly, *time-to-live* of packet-switching is roughly a special case of the *contract expiration* in contract-switching. Thus, contract-switching is a more general case of packet-switching with several additional flexibilities in terms of its economics and carefully reduced technical flexibilities due to scalability concerns particularly at the routing level.

Packet-switching introduced many more tussle points into the Internet architecture by breaking the *end-to-end circuits* of circuit-switching into *routable datagrams*. Contract-switching introduces even more tussle points at the edge/peering points of domain boundaries by *overlay contracts* as depicted in Figure 1.1.

Our research focuses on issues behind creating a contract-switching network architecture which allows flexible architecture involving financial and technical aspects so as to make guaranteed E2E QoS services available for the future Internet. We concentrate on the design of our contract-switching framework in the context of multi-domain QoS contracts. Our architecture allows such contracts to be dynamically composable across space (i.e., across ISPs) and time (i.e., over longer time-scales) in a fully decentralized manner. Once such elementary instruments are available and a method for determining their value is created (e.g., using secondary financial markets), ISPs can employ advanced pricing techniques for cost recovery and financial engineering techniques to manage risks in establishment of end-to-end contracts and performance guarantees for providers and users in specific market structures, e.g., oligopoly or monopoly. We build on top of our edge-based distributed dynamic capacity contracting (DDCC) framework [49], which was proposed for a single domain.

As DDCC can operate over ISP peering points, we employ contracts involving these ISP peering points and illustrate ways of realizing a contract-switched Internet core.

In particular, we investigate elementary QoS contracts and service abstractions at micro (i.e., *tens-of-minutes*) or macro (i.e., *hours or days*) time-scales. Measurement analysis on popular Internet destinations justify the efficacy of end-to-end guaranteed QoS services in macro time-scale in the sense that routes to these destinations are mostly stable for weeks [29, 33]. Although we believe that significant portion of value flows fit better in macro time-scale scheme, rising trends of on-demand and dynamic services require us to have micro time-scale operations in our architecture. We believe that traffic demands, which exhibit different temporal characteristics, will be best served in differentiated manner. For macro-level operation at high time-scales (i.e., *several hours or days*, potentially involving contracts among ISPs and end users), we envision a link-state like structure for computing end-to-end "contract routes." Similarly, to achieve micro-level operation with more flexibilities at lower time-scales (i.e., *tens-of-minutes*, mainly involving contracts among ISPs), we envision a BGP-style path-vector contract routing. Though there are similarities to QoS routing, the composition of contracts can involve multiple attributes, involve derivative contracts, and are based upon "contract-link-states" and "contract-path-vectors."

1.4 Organization of the Thesis

As the beginning part, we try to lay foundation for contract-switching architecture. On top this definition, we build our proposed "Contract Routing Framework" and implement "Link State Contract Routing Protocol (LSCR)" as a part of this framework. To support our proposed model, we implement LSCR Protocol in SSFNet

Framework [40] and evaluate our implementation on top of realistic network models. Our simulations with the real world Internet topologies show that, contract link definitions are robust against network load and topology changes upon drastic link failures. Also our simulations show that routing with contract links could be established on top of popular routing protocols like BGP and OSPF. In our work, we show that end-to-end QoS services could be achieved in LSCR scenario with market price stability and reasonable protocol overhead. Finally, our results reveal that duration of contract term plays a great role in protocol performance and market stabilization in contract-switched architecture.

Throughout this thesis, first, we give a brief introductory literature survey on inter-domain routing proposals in Chapter 2. In Chapter 3, our detailed architectural model is discussed. Next, Chapter 4 explains how Link State Contract Routing protocol is implemented and how LSCR implementation fits in the proposed architectural model. Performance evaluation of LSCR Protocol is given in Chapter 5. Finally, we have a discussion chapter where we summarize results and our contributions as well as our future work plan.

Chapter 2

Related Work

2.1 Inter-domain Routing Proposals

Inter-domain routing is a challenging research problem in the sense that it involves many facades including security, economics, reliability, service quality, scalability and more. As a result of it, many proposals have been made to target different sets of these issues but not all of them. So, it is hard to make a classification of these various research proposals. Yet, we want to group them in two different categories as improvement proposals on current architecture and clean slate approaches.

2.1.1 Improvement Proposals

Mahajan et al. [25] propose Nexit (Negotiated Exit) Framework for negotiating inter-domain paths taken by traffic flows originating from neighbor ISPs as shown in Figure 2.1. Nexit Framework is based on bilateral negotiation between directly connected ISP pairs who exchange their preferences on which traffic flow should take which inter-domain link connecting neighbors. Even for the cases where optimization criteria are

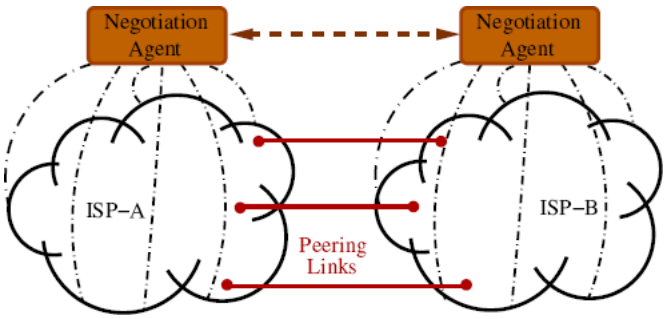


Figure 2.1: Path Trading Approach [25]

not compatible, both entities better off if they negotiate. (e.g ISP A minimizes delay whereas ISP B escapes overload.) Negotiation through preference exchange allows pairs to find better outcomes which are not explorable using default BGP mechanisms. The most exciting result of this work is that bilateral negotiation offers most benefits of global optimal routing without requiring ISPs to expose confidential their network topologies. Interesting enough, global optimal routing would end up in cases where one side of negotiation loses and the other side gains for sake of global optima whereas negotiating parties always end up in win-win or win-no-lose cases inherent to game theoretical approach. Another important result of this work is that cheating parties in negotiation do worse in compared to being truthful as in game theoretical repetition games where equilibrium is reached in a tit-for-tat fashion. Another similar work by Shavitt et al. [37], introduce term of bilateral "path trading" between neighboring ISPs in a bargaining problem scheme. As suggested by Nexit (and assumedly for path trading), a central entity negotiates with its neighbor on individual flow base over all traffic flows between neighboring entities. In contrast to that, contract-routing framework carries these proposed bilateral negotiations and preference exchange mechanisms into a generic multilateral scheme level where providers

exchange their preferred routes for downstream flows as contract link advertisements and contract path for upstream flows. According to our scheme, negotiations are spanned over time and multiple ISPs through market mechanisms instead of local consequential bargaining between neighbors. Results of these bargaining approaches are important since they point out that bilateral local improvements would have the most benefit of global optimal routing. In that sense, we can say that locality based Path-Vector Contract Routing will be able to find feasible paths as good as Link State Contract Routing with global perspective. In summary, path trading and negotiation proposals show us the hidden cost of shortest-path routing and the value of negotiation.

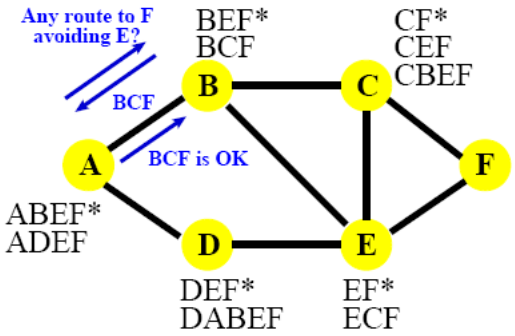


Figure 2.2: Multipath Routing [45]

Another backward compatible proposal with current Internet Architecture is Multi-path Inter-domain ROuting Protocol (MIRO) [45]. As default, providers learn inter-domain routes provided by BGP. To improve bandwidth or latency of default routes, or avoiding an intermediate ISP, source AS could initiate bilateral queries with intermediate ISPs along the default routes for alternative paths which are filtered due to policy or single shortest path constraint imposed by BGP. Bilateral path

queries are similar to those in above mentioned path trading proposals (see Figure 2.2). Moreover, MIRO extends bilateral negotiations by enabling negotiations with non-neighboring ISPs along the path. Alternative paths could be learnt through pull based queries at upstream as well as downstream AS could advertise alternative routes to upstream ASes in a push based manner for redirecting traffic along alternative paths for various purposes. (e.g. to avoid overload on default routes). Once alternative routes are learnt, necessary tunneling and state establishment are made upon initiator request. MIRO aims to leverage path diversity of Internet by bilateral path negotiations without state explosion risks and complete topology information requirements of router-level source routing schemes. Analysis results show that for discovering Internet path diversity MIRO could get most of the benefit that source routing would provide. Flexible structure which allows definition of policies in various granularities is also an advantage of MIRO. In compared to contract-routing, MIRO resembles a limited Path-Vector Contract Routing (PVCR) protocol where pull-push based queries made to explore alternative feasible paths. PVCR is more advantageous since source AS does not need to query all possible intermediate ASes instead initiated query will be propagated by willing ASes to established an end-to-end path establishment. Also in MIRO, source AS should monitor these alternative paths to capture their qualities. Without feedback of intermediate ASes, there is no guarantee that a better alternative path stays that way for a reasonable duration. So, even MIRO mitigates limitation on path diversity and single path constraints of current Internet architecture in a scalable way, it is not designed for neither providing guaranteed end-to-end services nor creation of a free market which could emerge on top of value-added connectivity services beyond best-effort services.

Another branch of improvement proposals can be classified under the umbrella

of "Service Overlay Networks". Basic idea behind these proposals is separation of forwarding and routing mechanisms. Cabernet [52], Routing as a Service [20] and Routing Control Platform [12] are some of the outstanding proposals aligned with this approach. Even though there are major differences between these research approaches, generally overall idea could be generalized as defining virtual link services on edge-to-edge connectivity capabilities of provider domains and stitching these virtual links with each other to compose end-to-end source routed paths. Contract Switching is also following very similar approach to define these virtual links and end-to-end contract path composition. Our contribution is introduction of innovative dynamic contracting mechanisms over these virtual links. Contract Routing approach automates current time-consuming, static SLA establishment process between service providers. In contrast to some of above proposals, Contract Routing does not propose any hard constraint or obstacle on emergence of Routing Service Providers aside from infrastructure owners (ISPs). Contract Routing approach emphasizes on differentiated pricing of point-to-point virtual links and introduce traffic differentiation according to their characterized life spans.

2.1.2 Clean Slate Approaches

Yang et al. [46] propose New Inter-domain Routing Architecture (NIRA). NIRA manages routing in three segments. Uphill segment is the path connecting source AS to the core of the Internet. In Internet core ASes have high connectivity degrees and they are densely connected with the others in Internet core by mesh like connection structures. Uphill segments are the cone-like regions of the Internet which consist of a provider on top, its customers and customers of its customers and so on. Within these uphill segments, an AS learn its provider and providers' provider and all transit paths

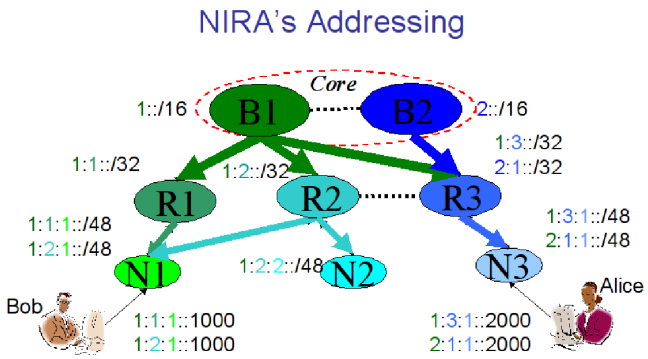


Figure 2.3: NIRA [46]

in its upgraph through topology information propagation protocol (TIPP). TIPP can advertise not only simple provider relationships but also dynamic link state updates on quality of these transit routes. NIRA does not intervene the routing processes in Internet Core. For downhill segment that connects Internet Core to destination AS, NIRA employs DNS like service named name-to-route lookup service (NRLS). NIRA employs provider rooted hierarchical addressing. A node address consists of two parts : 1) A prefix that is a non overlapping subdivision of provider address space 2) Provider independent intra-domain address part which uniquely identifies the node within intra-domain network (see Figure 2.3). First part does not only address a node but due to its hierarchical structure, it reflects the multi-domain AS level path to take to reach this destination. A multi-homed node would have multiple addresses in this scheme where prefix part of the address represents alternative uphill paths to this node and unique part uniquely identifies the node within specified domain

(e.g. 1:1:1::1000 1:2:1:1000). A user that wants to establish an end-to-end path, first choose an uphill path using information provided by TIPP, then lookup a downhill path by NRLS queries. Once such a path established, user makes use of source routing by adding source and destination addresses whose hierarchical addressing structure uniquely describes uphill and downhill paths this packet will take.

Hybrid Link-state Path-vector Protocol (HLP) [41] employs cone-like segmentation of end-to-end path establishment similar to NIRA. In this two-tiered model, segments representing cones are managed by link-state protocol within the cone. Link-state protocol localized within a cone allows keeping track of dynamic conditions of paths among providers within this cone. Between these hierarchical cones, a fragmented path vector (FPV) protocol manages the routing. Instead of announcing the whole intra-cone path to destination, FPV only announces the identifier of the cone, providers within the cone and the cost associated with paths leading to these provider domains. This two-tiered hierarchical routing model allows filtering of local topology changes if they do not cause any cost changes visible for the others outside the cone. (e.g. if there exists an equivalent cost path for replacement of failed path). In that sense, HLP reduces the number of route update messages significantly in compared to BGP. It provides isolation, localization of topology changes and linear time convergence capability. Currently, all of these issues compose a big threat for the scalability of the Internet.

Feedback Based Routing [51] is another proposal which separates route computation and forwarding plane from each other. ASes only exchange their inter-domain connections with each other. According to these topological information, each border router establishes a topology view of Internet and tries to compute two non-overlapping paths to each destination. Routers keep monitoring these paths by

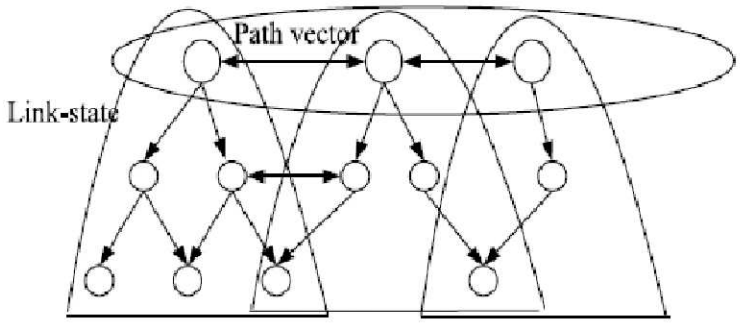


Figure 2.4: HLP [41]

means of Transmission Control Protocol (TCP) message sampling and Round Trip Time (RTT) analysis on them. One path serves as the back-up path so that once the packet transmission on active path failed, the back-up path takes over. Since the paths are computed in according to no interference rule, chance of the concurrent failure of both transmission path is minimized.

In compared to clean state proposals, Contract Routing does not require any hierarchical addressing scheme which entails huge changes on current Internet architecture. Instead Contract Routing can be built on current Internet architecture. Also as opposed to NIRA and HLP proposals where routing problem is fragmented in hierarchical or geographical segments, we believe that segmentation can be alternatively made on temporal basis according to traffic flow characteristics. More durable, long term traffic demand would be served specifically by a link-state protocol while more dynamic temporal demand would be best served by a path-vector protocol.

Chapter 3

Contract Routing Architecture

3.1 Contract Definition

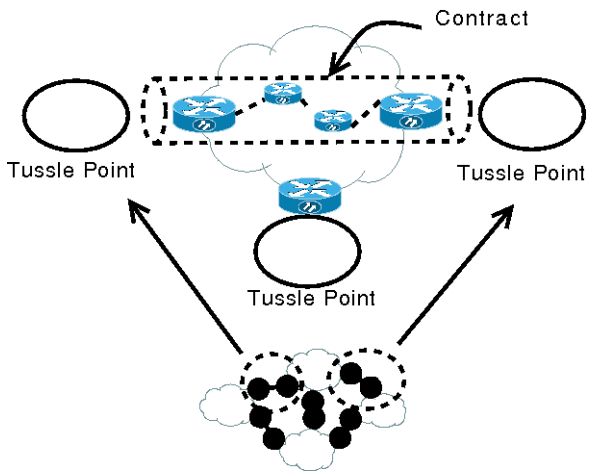


Figure 3.1: Contract Link Abstraction

Simply, "Contract Link" is a virtual link with an SLA. This virtual link abstraction represents routers, physical links, policies and all the required resources which at the end connect two edge routers of a provider domain (see Figure 3.1). Provided with an SLA, this edge-to-edge virtual link is a service definition made by

provider to advertise its edge-to-edge connection capabilities with QoS guarantees. While provider advertises service capabilities as contract link, contract link abstraction still allows provider to preserve and encapsulate its confidential network topology and business strategy in a competitive market. In such a market where providers advertise their edge-to-edge capabilities as contract links, each service providers become able to stitch these contract links to establish end-to-end "contract paths" with QoS guarantees. The key result of introducing such a scheme is that contract links bring *dynamic contracting* capability over peering points which is missing in current Internet today. Once provider domains are defined as set of contract links rather than points or hops in inter-domain routing problem, edge-to-edge services will become able to advertised with different prices in contrast to point-to-anywhere approach. It will surely require more complex pricing mechanisms where there can be $O(N^2)$ different prices instead of a single price for an ISP who has N peering points with its neighbors. Our research focuses on investigating complexity and feasibility of these economic models. As a final note on service provider classification in contract-routing architecture, there is no architectural constraints or hard coded separation of infrastructure operators (today's ISP) and pure contract switch service providers (routing service providers) brokering abstract services over resources of infrastructure operators as suggested by other proposals [20].

3.1.1 Contract Link Components

Although they can be extended to a larger set for further information exchange and flexibility, we define elementary components of contract links as 1) Time, 2) Financial and 3) Performance components in addition to definition of virtual path.

Virtual Link Description

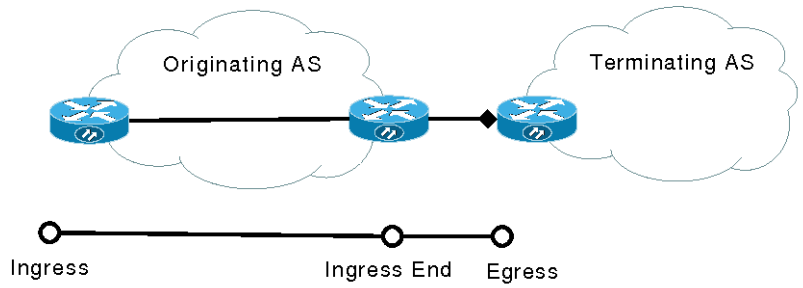


Figure 3.2: Contract Link Abstraction

Virtual link descriptor simply names the peering points of a provider domain involved in contract link advertisement. *Ingress Router* is the owner of the contract link advertisement. Once contract link is sold, ingress router creates a tunnel between ingress and egress peering points of the domain according to flow description. Created tunnel delivers encapsulated packets to the terminating AS. *Ingress End Router* is the egress peering point of the originating autonomous system (AS) where originating AS and terminating AS are connected with an inter-domain link. *Egress Router* is the point where virtual link terminates and packets belong traffic flow are decapsulated. In contract link definition, originating provider promises that it accepts packets from designated ingress router and deliver them to the terminating AS border router which is called egress router through the ingress end router exit of originating AS in according to terms of attached SLA (see Figure 3.2).

Performance Component

Performance Component defines network performance metrics for the virtual link. It may include metrics like bandwidth guarantees, packet drop ratio and availability. In addition to these items, delay, hop count, service level grades and many more

could be defined using this component. Promises made in performance component allows establishing end-to-end paths with guaranteed quality by stitching compatible contract links end-to-end. Additionally, performance component allows definition of void performance metrics where only bandwidth requirement is set for best-effort services in cases when the demanded service is not guaranteed QoS but avoiding an intermediate ISP on the path for security or financial concerns.

Time Component

As mentioned in Chapter 2, Contract Switching approach does rely on temporal differences in traffic demand characteristics while treating them. Exploitation of this separation within Contract Switching architecture makes time component one of the key figures in contract link definition.

Time component serves as a tool to describe several time related fields. One of them is *contract term* which defines maximum duration of the advertised service. Another field is named *offered after* which determines the earliest date that service subject to contract link will become available. A service provider using offered after field can advertise its services spread across a long time span towards future beginning with tens of minutes to maybe years as forward contracts and derivates so as to capture user demand and sell its products in priori.

This capability is more likely to help service providers to alleviate future unpredictability of market to a limit and hedge against the risks of the future. Also users have the choice of closing early deals for their future need of connectivity services now and guaranteeing their availability in the future.

In addition to these two elementary fields, many complementary fields could be added to allow more complex agreements.

Financial Component

Financial Component is the place where service providers express their price evaluation for their advertised service. Provided with time component, connectivity services can be advertised in various pricing schemes like spot pricing, forward contracts, options and many others. Inherent to guaranteed services, in case of unsuccessful delivery of these services it is required to define user compensation models within the umbrella of financial components. They may include money back guarantees or insurance terms. These insurance models not only assure user compensation but also provide market flexibility for providers in cases where delivering a service will become infeasible or even impossible.

3.1.2 Contract Link Types

Transit

Transit contract links are default type contract links which allows delivery of transit traffic through a provider domain.

Sink

Sink type contract links inform which ip prefix destination could be reached through which contract router. Sink type contract links represent virtual links connecting an ingress point to a subnet represented by an IP prefix.

Virtual

Virtual type contract links simply represent pure inter-domain links between stub and transit networks. Sink type contract links along with virtual contract links are

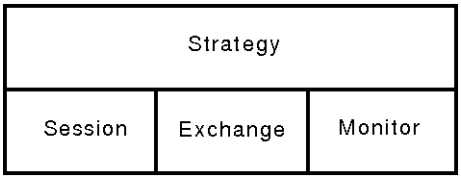


Figure 3.3: Contract Routing Framework

mostly necessary in case of pure contract switched architecture where there is no inter-domain routing protocol for providing best-effort connectivity services.

3.2 Architecture and Modules

3.2.1 Design Rationale

Contract routing architecture relies on modules and interfaces between module boundaries. This is a required model for next generation protocols since they should support independent upgrades and interplay of simultaneously running alternative protocols. Furthermore, their functions should support transparency and integration in case of third party involvement in monitoring, verification and authentication services. Beside these capabilities, architectural design should allow operators to escalate these functions to different network elements which carry tasks at different protocol layers as spanning aspects. Architectural design should avoid interfering with the choice of market players as much as possible.

3.2.2 Modular Design

As it also can be seen in Figure 3.3, following above principles, we define contract routing architecture in four modules as such: *Strategy:* Strategy module is the part where providers decide on following questions: How to utilize left-over capacity?

What should be the term for selling these resources? Should provider lock their resource in forward contracts or should it wait for selling on spot market? Should provider buy contract paths now by closing forward deals or should it wait for spot market? *Session:* which keeps track of established E2E Contract Paths. *Exchange:* which is responsible for exchange of contract link advertisements. *Monitor:* which responsible for monitoring of intra-domain resources for contract link establishments and contract paths by verifying QoS requirements on SLA conditions (Authentication and Authorization and Accounting).

3.2.3 Network Elements

Contract Routers reside at the edge of provider domain. Since G2G services within a domain may share physical network resources like routers and optical links, they need to be coordinated. However, it is provider discretion to whether or not to escalate this coordination task to network entities. At one extreme, ISP can create non-overlapping G2G services with minimum or no interference with each other using similar approaches described in [17] and let contract routers become independent market players according to designated ISP policy. There is no coordinator in this scheme. At another extreme, ISP deploys an operation and service support center (OSS) and has a monolithic network architecture. In this case, contract routers only advertise what OSS decides for them. Small scale providers may find these two approaches for their simplicity. In between these two, a provider may choose to employ multiple coordinators to manage its edge routers within independent sets and escalate necessary coordination tasks to responsible coordinators as depicted in Figure 3.4. This scheme more fits in large scale providers who have large number of contract links with diverse spatial characteristics.

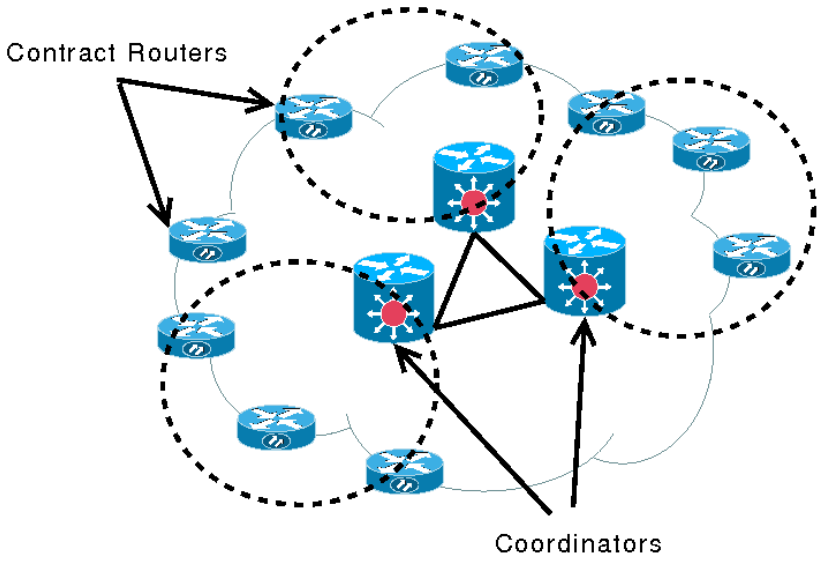
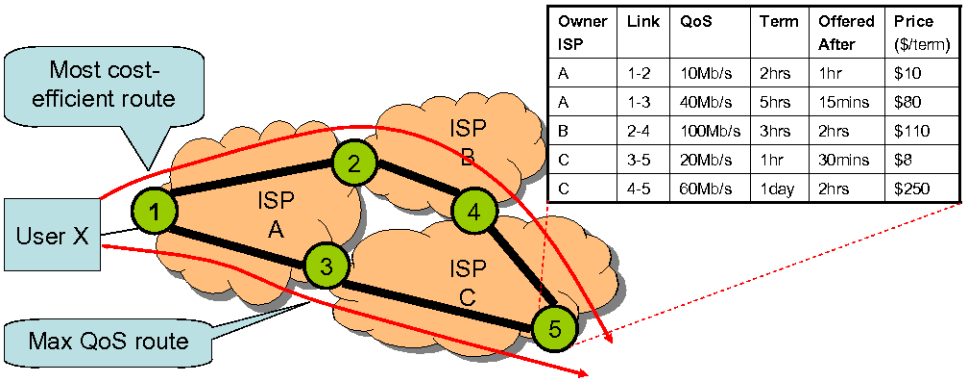


Figure 3.4: Network Elements

3.2.4 Proposed Design

Exchange Module

According to measurement analysis on popular prefix destinations, these routes are mostly stable for weeks [33]. So, these stable value flows could be served by macro time-scale (long term) contract paths whose terms span over time scales like (hours and days and longer). Although we believe that significant portion of value flows fit better in macro time-scale scheme, rising trends of on-demand and dynamic E2E services require us to define micro time-level schemes in our system. We believe that these operation schemes which have inherently different characteristics will be served best separately by simultaneously running protocols. In our framework, we target these macro time-scale operations by introducing a Link State Contract Routing (LSCR) protocol whereas micro time-scale (short term) value flows are targeted by Path Vector Contract Routing (PVCR) protocol as parts of exchange module.



Owner ISP	Link	QoS	Term	Offered After	Price ($/term)
A	1-2	10Mb/s	2hrs	1hr	$10
A	1-3	40Mb/s	5hrs	15mins	$80
B	2-4	100Mb/s	3hrs	2hrs	$110
C	3-5	20Mb/s	1hr	30mins	$8
C	4-5	60Mb/s	1day	2hrs	$250

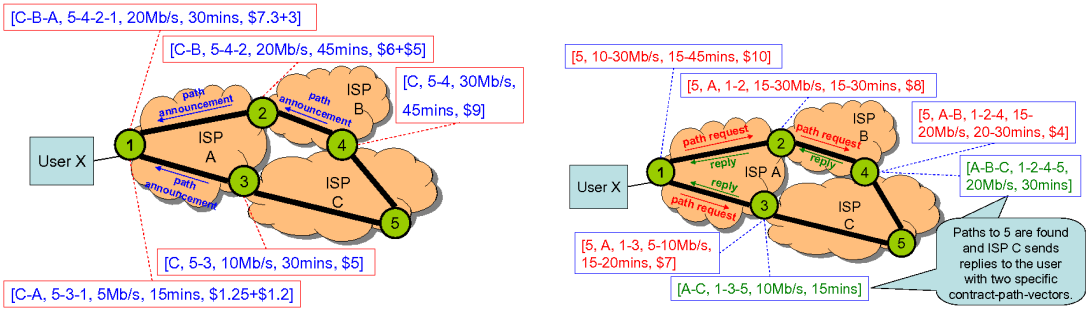
Figure 3.5: Scenario for LSCR

In LSCR protocol, contract advertisements are named contract link advertisements (CLA). Although their names resemble link state advertisements of OSPF, they should not be mistaken as frequently updated link state information since they represent SLA like contracts which attach financial and technical obligations. Even it looks prohibitive to run a link state protocol in inter-domain area, we believe that macro time-scale contract terms, careful filtering and economies of scale principles let convergence and scalability characteristics of LSCR fit well in this scheme.

While LSCR allows composition of globally optimal contract paths according to complete topology view, PVCR allows online query and on the fly composition of contract paths upon provider initiation (or on user demand) with local capabilities. So, both of these protocols allow us to exploit different characteristics of path-vector and link-state protocols so as to cover different classes of traffic demand.

Session Module

Contract link represents a tunnel as depicted in Figure 3.2. So, once a contract link is sold, promised tunnel should be established between ingress router and egress router.



(a) Lazy calculation (b) On-demand reactive calculation

Figure 3.6: Path-vector contract routing: (a) *Provider initiates*. (b) *User initiates*

Session Module simply takes care of establishment and keeping track of these tunnels.

Monitoring Module

Contract routing brings well-defined provider compensation mechanisms as well as accountability in provided services. But these capabilities require deployment of monitoring tools for both consistency of advertised services as well as verification of established contract paths. Security tasks also should be carried within this module. We envision third party trustee mechanisms (as in the case of credit card transactions) to authenticate and verify contract links.

3.3 Path Vector Contract Routing

To provide enough flexibility capturing more dynamic technical and economical behaviors in the network, it is possible to design contract routing that operates at short time-scales, i.e., *tens of minutes*. This time-scale is reasonable as current inter-domain BGP routing operates with prefix changes and route updates occurring at the order of a few minutes [42]. Further, an ISP might want to advertise a spot price for

an edge-to-edge contract to a subset of other ISPs and Contract-Switched Network Providers (CSNP) instead of flooding it to all. Similarly, a user might want to query a specific contracting capability for short-term and involving various policy factors. Such *on-demand reactive* requests cannot be adequately addressed by the link-state contract routing.

Just like BGP composes paths, E2E contract paths can be calculated in an on-demand lazy manner. In our design, each ISP has the option of initiating contract path calculations by advertising its contract links to its neighbors. Depending on various technical, financial, or policy factors, those neighbors may or may not use these contracts in composing a two-hop contract path. If they do, then they advertise a two-hop contract path to their neighbors. This path-vector composition process continues as long as there are participating ISPs into the contract paths. Users or ISPs receiving these contract paths will have the choice of using them or leaving them to invalidation by the time the contract path term expires.

Provider Initiates: Figure 3.6(a) shows an example scenario where a provider initiates contract-path-vector calculation. ISP C announces two short term contract-path-vectors at peering points 3 and 4. The ISPs B and A decides whether or not to participate in these contract-path-vectors, possibly with additional constraints. For example, ISP B reduces the capacity of the initial path-vector to 20Mb/s and increases its price to \$11. Though each ISP can apply various price calculations, in this example ISP B adds \$5 for its own contract link 2-4 on top of the price of the corresponding portion (i.e., \$9*20/30 = \$6) of the contract link 4-5 coming from ISP C. Similarly, ISP A constrains the two contract-path-vector announcements from ISPs B and C at peering points 2 and 3 respectively. Then, the CSNP (or ISP A) offers the two contract-path-vectors to the user X, who may choose to use the 1-5 short-term

QoS path. In this path-vector computation scheme, whenever an ISP participates in a contract it will have to commit the resources needed for it, so that the users receiving the contract path announcements will have assurance in the end-to-end contract. Therefore, ISPs will have to decide carefully as potential security and trust issues will play a role. This game theoretic design exists in the current BGP inter-domain routing. In BGP, each ISP decides which route announcements to accept for composing its routes depending on policy, trust, and technical performance.

User Initiates: Users may query for an E2E short-term contract path with specific QoS parameters which do not exist in the currently available path-vector. This kind of design can potentially allow involvement of end users into the process depending on the application-specific needs. For example, in Figure 3.6(b), user X initiates a path-vector calculation by broadcasting a "contract-path request" to destination 5 with a capacity range 10-30Mb/s, term range 15-45mins with up to $10 of total cost.

This contract-path request gets forwarded along the peering points where participating ISPs add more constraints to the initial constraints identified by the user X. For example, ISP B narrows the term range from 15-30mins to 20-30mins and the capacity range from 15-30Mb/s to 15-20Mb/s while deducting $4 for the 2-4 contract link of its own from the leftover budget of $8. Such participating middle ISPs have to apply various strategies in identifying the way they participate in these path-vector calculations. Once ISP C receives the contract-path requests, it sends a reply back to user X with specific contract-path-vectors. The user X then may choose to buy these contracts from 1 to 5 and necessary reservations will be done through more signaling.

3.4 Link State Contract Routing

One version of inter-domain contract routing is link-state style with long-term (i.e., *hours or days*) contract links. For each contract link, the ISP creates a "contract-link-state" including various fields. We suggest that the major field of a contract-link-state is the forward prices (or prices committed for a later deal) in the future as predicted by the ISP now (based upon anticipated future loads). Such contract-link states are flooded to other ISPs and CSNPs. Each ISP will be responsible for its flooded contract-link-state and therefore will have to be *proactively* measuring validity of its contract-link-state. This is very similar to the periodic HELLO exchanges among the routers in an OSPF domain. When remote ISPs obtain the flooded contract-link-states, they can offer point-to-point and end-to-end contracts that may cross multiple peering points. Though link-state routing was proposed in an inter-domain context [9], our "contract links" are between peering points *within* an ISP, and not between ISPs (see Figure 3.5).

To compute the end-to-end "contract paths", the local agent of CSNPs or ISPs performs a QoS-routing like computation procedure to come up with source routes, and initiates a signaling protocol to reserve these contracts.

Figure 3.5 shows a sample scenario where link-state contract routing takes place. There are three ISPs participating with 5 peering points.

For the sake of example, a contract-link-state includes six fields: *Owner ISP*, *Link*, *Term* (i.e., the length of the offered contract link), *Offered After* (i.e., when the contract link will be available for use), and *Price* (i.e., the aggregate price of the contract link including the whole term). ISPs have the option of advertising by flooding their contract-link-states among their peering points. Each ISP has to

maintain a contract-link-state routing table as shown in the figure. Some of the contract-link-states will diminish by time, e.g., the link 1-3 offered by ISP A will be omitted from contract routing tables after 5hrs and 15mins. Given such a contract routing table, computation of "shortest" QoS contracts involves various financial and technical decisions. Let's say that the user X (which can be another ISP, CSNP, or a network entity having an immediate access to the peering point 1 of ISP A) wants to purchase a QoS contract from 1 to 5. The CSNP can offer various "shortest" QoS contracts. For example, the route 1-2-4-5 is the most cost-efficient contract path (i.e. (10Mb/s*2hrs + 100Mb/s*3hrs + 60Mb/s*24)/($10 + $110 + $250) = 27.2Mb/s*hr/$), while the 1-3-5 route is better in terms of QoS. ISPs can factor in their financial goals when calculating these "shortest" QoS contract paths. The 1-2-4-5 route gives a maximum of 10Mb/s QoS offering capability from 1 to 5, and thus the CSNP/ISP will have to sell the other purchased contracts as part of other end-to-end contracts or negotiate with each individual contract link owner. Similarly, the user X tries to maximize its goals by selecting one of the offered QoS contracts to purchase from 1 to 5. Let's say that the CSNP/ISP offers user X two options as: (i) using the route 1-2-4-5 with 10Mb/s capacity, 2hrs term, starting in 5hrs with a price $15 and (ii) using the route 1-3-5 with 20Mb/s capacity, 1hr term, starting in 30mins with a price $6. Let's say that user X selects the 1-3-5 route. Then, the CSNP/ISP starts a signaling protocol to reserve the 1-3 and the 3-5 contract links, and triggers the flooding of contract link updates indicating the changes in the contract routing tables.

One issue that will arise if an ISP participates in many peering points is the explosion in the number of "contract links", which will trigger more flooding messages into the link-state routing. But, the number of contract links can be controlled by

various scaling techniques, such as focussing only on the longer-term contracts offered between the major peering points and aggregating contract-link-states as region-to-region where a region corresponds to a set of peering points. Also, a key difference between our proposed link-state contract routing and the traditional intra-domain link-state routing is that *floods only need to be performed if there is a significant change on contracting terms or in the internal ISP network conditions.*

However, in traditional link-state routing, link-states are flooded periodically regardless if any change has happened.

Chapter 4

LSCR and SSFNet Contribution

For this thesis work, we set the general layout for contract-routing framework. As a part of this framework, we implement Link State Contract Routing (LSCR) protocol and leave the implementation of Path Vector Contract Routing (PVCR) protocol for the future work. Since our target is to build contract links as overlays on top of provider intra-domain topologies, we need a network simulator which provides us today's popular network protocol implementations currently used by service providers. Since SSFNet Framework [40] provides us OSPFv2 [28], BGP4 [32], TCP and IPv4 protocol implementations, we choose SSFNet Network Simulator to build our framework on it. SSFNet framework is implemented in Java Programming Language. For this reason, we also develop LSCR protocol in Java. For the following sections, first our contributions for SSFNet OSPFv2 implementation are shared in short and then, LSCR implementation will be given in detail.

4.1 Improvements on SSFNet OSPFv2 Implementation

OSPFv2 protocol implementation of SSFNet leaves out AS-External LSA capabilities. AS-External LSA advertises IP prefixes external to provider domain. Without having AS-External LSA capabilities, routers within the domain would simply drop packets destined to external IP prefixes [28]. Although static routes could be configured on core routers to prevent packets from being dropped, it is a static approach and static approaches are weak against link failures and topology changes. Another way of handling this task is connecting edge routers with direct links so as to rely on internal and external BGP capabilities to carry packets to foreign prefixes. But this approach significantly limits the realistic modeling capabilities of our simulations since with this scenario traffic flows destined to external prefixes just simply follow these direct links between edge routers and avoid provider core routers.

To avoid above pitfalls and to be able to have realistic intra-domain topologies in our simulations, we implement OSPFv2 AS-External LSA mechanisms described by Section 4, 12.2.4, 16.4 and other relevant sections of OSPFv2 Request For Comment (RFC) 2328 [28] for calculating AS external routes, creating and flooding AS-External LSAs. In SSFNet framework, within OSPF protocol IP forwarding table listeners, we implemented mechanisms designed for border routers so as to keep track of external prefixes added or removed by BGP protocol. AreaData structure is also revised to include external route in path calculation process. Whenever cost of external path or path to an external route changes, we flood these changes to core routers of provider domain through AS-External LSA updates. Interested readers may find the complete list of changes and actual implementation in our web site [39].

4.2 Link State Contract Routing Protocol

4.2.1 Network Architecture

Contract Routing Framework does not impose any particular network architecture on service providers as long as proposed functions and modules are implemented as configurable and accessible services in compliance with the protocol definition. Having said that, in our implementation of LSCR we prefer to have a monolithic network management model for provider domain for its simplicity. In this simple model, there is a contract router at each peering point of provider domain and all of these contract routers are managed by a single central coordinator (or "Central Broker" as we will call it through the text). For our particular design choice, we assign most of the monitoring and strategy tasks to our Central Broker as a result of monolithic network management approach. General operations and structure of our architecture are given in Figure 4.1. In short, strategic capacity assignment and pricing procedure where provision of common network resources between overlapping edge-to-edge services are made, tunnel and session establishment of sold contract links, verification of QoS promises made by advertised contracts can be listed as some of the important tasks carried by Central Broker. In this implementation, contract routers are kept pretty simple. Contract routers are simply responsible for advertising instructed contract links and establishing end-to-end contract paths in according to traffic demand model projection of Central Broker. In our particular implementation, due to our promises of being practical and being compatible with current Internet architecture, contract routing protocol simply sits on existing BGP and OSPF protocols. According to that, a contract router could be defined as autonomous system border router which have running BGP and OSPF protocol sessions on it.

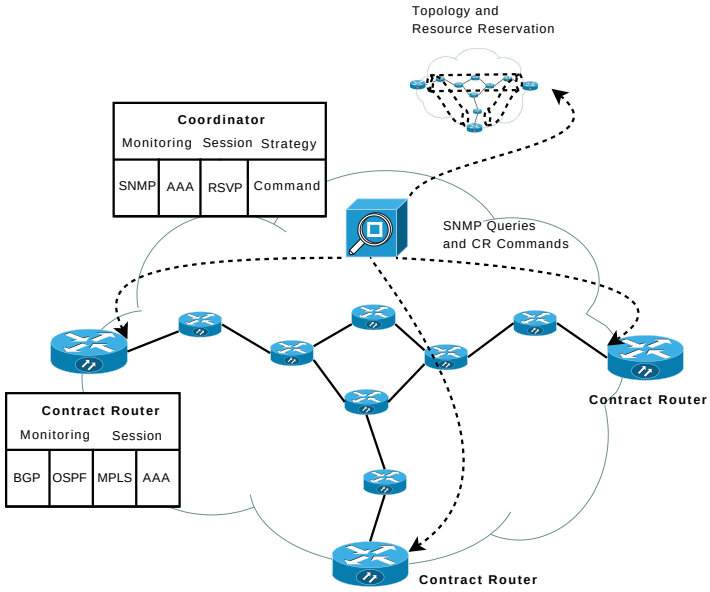


Figure 4.1: General View of our Contract Routing Implementation

4.2.2 Central Broker

Object Descriptions and Data Structures

Topology Monitor: Topology Monitor is the holder for a set of function implementations under Monitoring Module of Contract Routing Architecture. Basically it invokes related functions in case of link failures and topology changes. It consists of a timer which triggers link failure probation of contract routers regularly. Functions here mentioned usually handled by Simple Network Management Protocol (SNMP) monitors, traps and alarms associated with critical resources in a reactive manner where network entities inform operation support unit. Since we do not have these SNMP capabilities on SSFNet framework, we emulate these capabilities by introducing a topology monitor who regularly probes contract routers for discovering topology

changes in a proactive manner.

Traffic Demand Modeler: Traffic Demand Modeler is the holder for functions which provide customer traffic demand projections for destinations outside the provider domain.

Strategy Engine: Strategy Engine is a decision maker whose functions are regularly invoked in provider decision cycles. According to provided policies, Strategy Engine generates resource allocation, provisioning, pricing and capacity management strategies and commands when invoked by Central Broker.

Edge-to-Edge Path Manager: This holder simply keeps the list of links and routers along edge-to-edge shortest paths connecting contract router pairs of provider domain. QoS related statistics regarding bandwidth capacity, delay, packet drop ratio, and packet queue statistics along the way are held in this data structure. In our implementation, we limit these statistics to bandwidth capacity of G2G path since we only target to investigate bandwidth guaranteed services in our simulations for this thesis work, however, by deploying Traffic Engineering Extensions for OSPF protocol (OSPF-T), today providers can easily monitor their networks for above mentioned QoS performance metrics.

Traffic Flow Holder: Traffic Flow Holder stores the list of advertised and sold contract links. There can be found contract link descriptors and related statistics in this data structure.

Functions

Capacity Monitoring and Verification Central Broker should know edge-to-edge bandwidth capacity of paths between contract routers to be able to offer bandwidth

guaranteed edge-to-edge services. For carrying this task, central broker periodically retrieves information on shortest paths connecting contract routers by probing OSPF sessions running on contract routers. OSPF protocol AreaData structure keeps periodically refreshed shortest path tree connecting the router to the other router which runs OSPF protocol.

Shortest path calculation is made according to Dijkstra's Shortest Path Algorithm where each link is associated with a value representing the OSPF distance (or cost). Usually it is a common practice to assign reciprocal of link bandwidth capacity as link's OSPF distance. According to this, shortest path is probably the widest path which have higher end-to-end bandwidth capacity with the least hop along the way or a close intermediate between these two.

Shortest path (SP) tree representation consists of a list of interface pairs of neighboring OSPF routers. By matching its link inventory database with interface pairs given by SP, central broker gathers bandwidth capacity information of edge-to-edge directional SP connecting two arbitrary contract routers. Aftermath of a link failure or a topology change, these calculated shortest paths may change. So, central broker periodically keeps track of edge-to-edge shortest paths and their bandwidth capacities.

We use Java object reflection mechanisms of SSFNet framework to retrieve shortest path tree of OSPF sessions. In reality this data structure can be accessed through SNMP MiB for OSPFv2 remotely or open interfaces provided by product specific protocol API locally.

Session Management Once a contract link is sold, required G2G tunnel represented by contract link should be established using IP tunneling, label tunneling or by any other means of tunneling technologies. Required network resources should be

dedicated to ensure delivery of guaranteed QoS promises. These dedicated resources should be released when the session expires. In our implementation, central broker keeps track of creation and termination of sessions through session management timers and structures. Also, by using verification information provided by capacity monitoring and verification functions, it verifies if underlying edge-to-edge path still meets the QoS promises made by sold contract links. Today's routers have several SLA monitoring mechanisms which includes sending periodic ping messages, counting and monitoring TCP sessions and many others for several performance metrics. In our implementation, we only use bandwidth capacity monitoring for verification.

Traffic Demand Projection In our architecture, a service provider has a two sided task. First of all, as a service provider, it advertises contract links. In the second place, it buys contract links advertised by other service providers to establish end-to-end QoS guaranteed paths requested by customers of its domain. As a part of later task, service provider should decide on which contract paths are to be established to satisfy customers' demand on end-to-end QoS services. In our implementation, we allow several generation method for traffic demand projection values. Yet in our simulations, we only deploy static customer demand calculated in according to realistic gravity based methods described in Chapter 5. We think that at this stage of our research, it is sufficient to work with static demand since it is a different problem by itself to estimate traffic demand matrices by traffic sampling and user modeling. To summarize traffic demand projection generation, Central Broker queries Traffic Demand Modeler for external IP prefixes and demand for bandwidth guaranteed service for these IP destinations. Then, Central Broker instructs contract routers to establish contract paths to satisfy projected customer demand.

Provision There are several tasks carried by provision function. These could

be listed as follows:

- How to allocate common network resources such as link bandwidth capacity in case of overlapping edge-to-edge shortest paths between contract routers?
- How to price and capacitate (in terms of guaranteed bandwidth promises) contract link advertisements?
- How to set duration of contract link advertisements as binding contract terms?
- In case of a drastic topology change, how to decide which contracts to be bailout?

For the first item, we deploy fair-share capacity distribution among edge-to-edge services as follows:

- Find the minimum link capacity as bottleneck link along edge-to-edge path for each edge-to-edge path
- Assign a drop capacity (for our case it is 100 Mbps)
- Add drop capacity to each edge-to-edge path's bucket in turn, until all resource capacity is assigned or all bucket capacities reach to their bottleneck link capacity associated with edge-to-edge path.
- In case of excess capacity left, distribute this excessive capacity equally among edge-to-edge paths.

Above mechanism is implemented as a heuristic to replace Linear Programming counterpart which would be costly to call frequently.

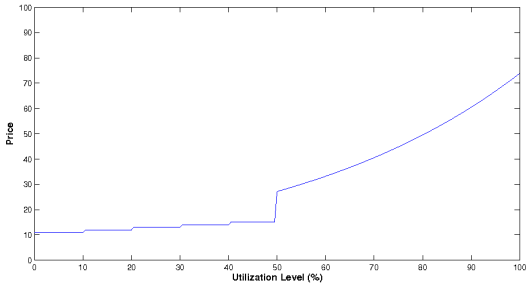


Figure 4.2: Strategy Engine Function: Pricing Function

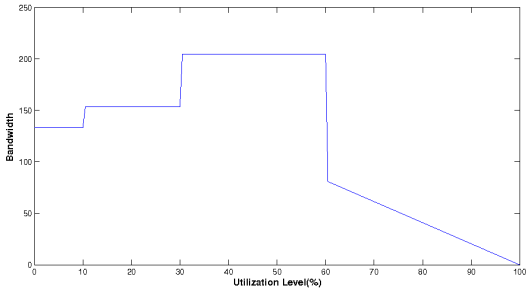


Figure 4.3: Strategy Engine Function: Capacity Function

For pricing and capacitating contract link advertisements, we deploy simple functions that deploy edge-to-edge path capacity utilization moving average as a parameter.

As seen in Figure 4.2, pricing function is a step function increasing by incremental steps for utilization levels below 60 percent to indulge utilization level below that level. For higher utilization levels, exponential increments punish utilization levels close to high load. Similar strategy can be observed also for bandwidth capacity assignment function behavior plotted for a sample 1 Gbps edge-to-edge path in Figure 4.3. Bandwidth capacity assignments increase up to 50 percent utilization level, then decrease inversely with increasing utilization.

For the contract term, we have set various durations starting with 10 minutes increasing up to 1 week as a simulation parameter. As our simulation results reveal that duration of contract term has a profound impact on routing performance and market conditions in such a contract-switched architecture.

Timers

Startup Timer: This timer initiates bootstrap process for Central Broker.

Session Expiration Timer: This timer is a part of session termination process of sold contract links.

Advertisement Strategy Timer: This timer calls related Strategy Engine functions to make it reevaluate feasibility of contract link advertisements upon a significant capacity change of an edge-to-edge path due to topology change or reserved bandwidth capacity for sold contract links.

Capacity Monitoring Timer:

Retransmission Timer: Since LSCR protocol employs Unreliable Datagram Protocol (UDP) encapsulated messages for transport, retransmission of dropped or corrupted messages is necessary for a reliable transport. Retransmission timer simply resend the packets in retransmission lists.

Configuration Parameters

Network Model Type An AS can be configured as STUB network which does not allow any advertisement of contract links. Contract routers in a STUB network only play a customer role in Contract Switched Internet so as to meet their domain's

customers' demand on E2E QoS guaranteed services. If an AS is configured as a TRANSIT NETWORK contract routers both advertise contract link advertisement as service provider and establish contract paths as service whole-sellers to their customers.

Topology Monitor Probe Interval This parameter determines how frequent central broker proactively monitors topology changes. In our model we set this parameter as 3 minutes due to BGP Keep Alive Timer mechanisms. According to that if BGP Session does not receive 3 consecutive keep-alive messages within 180 seconds then BGP Session resets its connection with its peer. Within 3 minutes, we can capture link failures on inter-domain links connecting AS with its neighbors. Since OSPF reacts to topology changes in 15 minutes, 3 minute probe interval can successfully capture intra-domain topology changes easily.

Advertisement Strategy Update Invocation Thresholds Due to system stability and prevalence of contract link advertisements, central broker could not update contract link advertisements frequently. Only major topology changes or utilization level changes could lead to updates. Update Invocation Thresholds represent the minimum amount of change which is high enough to invoke update mechanisms. For our work, we have two types of thresholds : 1) "Timed Threshold" and 2) "Utilization Percentage Threshold". Updates on a contract link advertisement will be suppressed during Timed Threshold if update mechanisms are called multiple times during that period. If edge-to-edge capacity changes below Percentage Threshold, contract link advertisements related to that edge-to-edge path will be suppressed also.

Strategy Engine Parameters Base Price is the initial price requested for per bandwidth per second by an ISP, Base Bandwidth is the minimum amount of bandwidth level advertised by an contract link advertisements.

Moving Average Parameter Since terms of contract link advertisements are determined by functions which are driven by edge-to-edge capacity utilization, drastic changes in utilization levels would lead system instability. To prevent this, we feed long term average of utilization values to functions depicted in Figure 4.3 and 4.2.

4.2.3 Contract Router

Object Descriptions and Data Structures

Neighbor Neighbor is the representation of peer contract router of a neighbor ISP. Since link state protocols require all neighbors to have full synchronization of routing information, neighbor state is important. DOWN, EXCHANGE, LOADING and FULL are the states one neighbor could be in. In exchange state, peers exchange their contract link advertisement database summaries. In loading stage, neighbors requests the contract links either they do not have or they have an outdated version. In full stage, both neighbors have the same inventory of contract link advertisements. From this point on, peering between neighbors are fully established and all newly arrived updates on a contract router will be flooded to peer contract router. Neighbor data structure represents the state machine which handles above described exchange and synchronization functions.

Interface Interfaces are abstract representatives for the physical interfaces of contract routers and also holder data structures for neighbors. An interface can have multiple neighbors. Interfaces are uniquely identified by IP addresses.

Contract Link Database Contract Link Database holds contract link advertisements both received from neighbors and also advertised by contract router itself. It

is also responsible for the periodic aging of the contract link advertisements.

Path Calculator Path calculator is Path Calculation Engine which calculates end-to-end paths by stitching contract link advertisements. In our framework, we implement four variations of Dijkstra's Shortest Path Algorithm to calculate shortest path in according to different criteria. Widest Shortest Path variation simply calculates the shortest path with maximum bandwidth capacity. Cheapest Shortest Path variation selects the shortest of the cheapest end-to-end paths. Other two variations are similar shortest path calculation methods with minimum bandwidth limit (Cheapest Shortest Path with minimum bandwidth requirement) and maximum budget limit (Widest Shortest Path with budget constraint).

Contract Path Information Base Paths calculated by Path Calculator are cached at Contract Path Information Base (CPIB). In our model, for simulation purposes end-to-end paths to selected prefix destinations are calculated and kept at CPIB. Path calculator is invoked regularly and entries of CPIB are kept updated. Path calculator is invoked upon Contract Link Database changes either due to updates or withdrawals of contract link advertisements. For other purposes, CPIB could be limited or filtered to some subset of prefix destinations and are not required to be up-to-date so as to keep computation and memory burden low.

Contract Manager Contract Manager is the proxy between Central Broker and Contract Router. Central Broker commands contract router to establish end-to-end bandwidth guaranteed paths described by destination IP prefix and required guaranteed bandwidth levels. Contract Manager checks the Contract Path Information Base for calculated paths. If there exists a contract path which meets the QoS requirements and budget constraints, then contract manager initiates Resource Reservation Protocol (RSVP) signaling to establish this end-to-end path. Contract manager establishes,

monitors and terminates these established contract paths.

RSVP Session Manager All RSVP signaling procedures are handled by RSVP session manager. RSVP Path establishment, also tear down mechanisms and message types are similar to their counterparts given in RSVP protocol described by RFC 2205 [7]. In that scheme, sending an RSVP Path Teardown message means bailing out of the established contract path. Also tunnels are monitored through RSVP periodic refresh messages.

Contract Path Forwarding Table When end-to-end contract paths are established through RSVP signaling among contract routers, contract routers establish the edge-to-edge tunnels promised by contract link advertisements. Then, they install these tunneling entries into Contract Path Forwarding Table which underlying tunneling protocols (e.g. Multiprotocol Label Switching (MPLS) [6, 11] or IP in IP encapsulation [31]) will use for actual packet forwarding.

Functions

Exchange Protocol

Path Computation

Session and Tunneling Management

Contract Path Acquisition

Timers

Startup Timer Bootstrap timer for contract routing session.

Aging Timer Aging timer regularly increments the age of contract link advertisements on contract link database. If the age of a contract link advertisement hits

the maximum allowed age by the protocol, then it will be removed from the database and expired contract link advertisement will be flooded to neighbors so as to make them flush down this particular contract link from their databases.

Route Calculation Timer Since multiple contract link update messages may coincide within a short time interval, it is necessary to suspend route calculation invocation on updated contract link database for a period of time to avoid cascading updates. Route Calculation Timer waits during the SPFHOLDTIME upon a change on contract link database before it invokes route calculation.

Periodic Trace Timer Periodic Trace Timer is a simple monitor which sends and receives back trace messages so as to examine end-to-end path followed by tunneled packets for QoS verification.

Configuration Parameters

Budget Contract Manager has an initial budget in terms of credits which are spent on establishment of contract paths. These credits are reimbursed upon the termination of contract path. So, contract manager can not pay extreme prices beyond its budget.

Proactive Contracting Parameters To assure uninterrupted QoS services, contract router has to react before contract path termination. Proactive contracting parameters simply determine when contract router should react to replace soon to be expired contract paths.

Traffic Demand Multiplier Traffic demand parameter is a simulation parameter which determines the multiplier for central broker to magnify traffic demand so as to simulate high and moderate load cases.

Startup Delay Parameters Contract Router, RSVP Session Manager, Path Calculator and Contract Manager have separate startup delay parameters for their bootstrap periods.

Strategy and Path Calculation Parameters Among four path calculation method, contract routers choose one to apply for their market strategy. According to that, a player either targets to acquire widest, cheapest end-to-end paths or their balanced versions with bandwidth requirements or budget constraints. Path calculation suppression timer parameters like SPFHOLDTIME also can be listed here.

Chapter 5

Simulations

5.1 Setup

It is a well-known research problem to simulate Internet in a realistic manner [30]. There are several contributors to this well-known problem. First of all, due to the scale of Internet capturing realistic topology *(or map of an Internet)* is a big challenge. Although there are several proposals which include topology sampling of Internet topology maps [19, 24], synthetic topology generation by imitating several characteristics of Internet [8, 27, 44] and several topology mapping approaches in different granularities [36,38], problems of capturing Internet topology characteristics and generating a realistic topology at a manageable scale for simulation purposes are not yet solved in practical. Due to complexity and scale of the problem, modeling approaches target either to cover some specific characteristics of Internet (e.g. clustering coefficient, degree distribution) or to provide topology information (captured or emulated) for a particular granularity (e.g. domain level).

In Contract Routing Framework, provider domains are not just single hops

at inter-domain level but sets of virtual links representing their intra-domain infrastructure. So, according to this model we need both realistic intra-domain and inter-domain topology maps for our simulations. At this point, we choose to have realistic intra-domain topology and have limited in size inter-domain topology map. Considering large scale of a common level-1 ISP intra-domain topology map (>100 routers), we have to make a compromise in the scale of inter-domain level topology for our simulations. This will limit the number of ISPs in our inter-domain level topology at a level of <20.

As a second problem, a reasonable simulation model requires us to have a realistic traffic demand model close enough to test our protocol successfully.

For the following sections, we try to describe our approaches to tackle above mentioned issues to have a realistic simulation model.

5.1.1 Intra-domain Topology

For our intra-domain topology model, we use router level intra-domain topologies of 6 ISPs *Telstra, Sprint, Exodus, Tiscali, Abovenet and Telstra* provided by Rocketfuel Topology Maps [38]. Rocketfuel Topology Maps provide us the adjacency and link propagation matrices and also link weights of these mapped topologies. So, we still need to model link capacities and estimate traffic demand. Below steps summarize our methodologies:

BFS Based Router Classification

A provider domain can be basically described as a set of backbone and edge routers. Backbone routers are the ones connected by high capacity long-haul links in the backbone side of the network away from the last mile. Edge routers mostly reside at

Table 5.1: Rocketfuel-based router-level ISP topologies.

ISP	# of Routers	# of Links	Degree (avg/max)	BFS Distance (avg/max)
Abovenet	141	922	6.6/20	2.3/4
Ebone	87	404	4.7/11	3.3/7
Exodus	79	352	4.5/12	3.0/5
Sprintlink	315	2334	7.4/45	2.7/7
Telstra	108	370	3.8/19	3.5/6
Tiscali	161	876	5.6/31	2.6/5

presence points of the domain closer to cities where actual traffic flows are originated and terminated. So, we need to make a classification of edge and backbone routers first for traffic modeling and link capacity assignment.

We do the following procedure:

1. Tag the most connected router of the domain as the center
2. Do a BFS traversal on topology as rooted from the center of domain
3. Assess BFS distance and node connectivity degrees and identify *Degree Threshold* and *Distance Threshold* so that edge routers correspond to 75-80% of all nodes in topology
4. Tag routers with node degrees less than *Degree Threshold* and DFS distances greater than *Distance Threshold* as *edge routers*.

Link Capacity Assignment

In order to assign estimated capacity values for individual links of the Rocketfuel's topologies, we use a technique based on the Breadth-First Search (BFS) algorithm. We, first, select the maximum-degree router in the topology as the center node for

Table 5.2: Rocketfuel-based router-level ISP topologies.

ISP	Degree Threshold	BFS Distance Threshold	# of Edge Routers	# of G2G Flows
Abovenet	9	3	108	11,556
Ebone	6	4	66	4,290
Exodus	6	4	60	3,540
Sprintlink	9	5	254	64,262
Telstra	5	4	84	6,972
Tiscali	8	4	125	15,500

BFS to start from. After running BFS from the max-degree router, each router is assigned a *BFS distance* value with respect to the center node. The center node's distance value is 0.

Given these BFS distances, we apply a very simple strategy to assign link capacities: Let the BFS distances for routers i and j be d_i and d_j respectively. For the links (i, j) and (j, i) between the routers i and j, the estimated capacity $C_{i,j} = C_{j,i} = \kappa[\max(d_i, d_j)]$ where κ is a decreasing vector of conventional link capacities. In this paper, we used: $\kappa[1] = 40Gb/s$, $\kappa[2] = 10Gb/s$, $\kappa[3] = 2.5Gb/s$, $\kappa[4] = 620Mb/s$, $\kappa[5] = 155Mb/s$, $\kappa[6] = 45Mb/s$, and $\kappa[7] = 10Mb/s$. So, for example, a link between the center router and a router with BFS distance 5 will be assigned 155Mb/s as its estimated link capacity. Similarly, a link between routers with distances 1 and 3 will be assigned with a capacity estimation of 2.5Gb/s. The intuition behind this BFS-based method is that an ISP's network would have higher capacity and higher degree links towards center of its topology. This intuition is well-supported by the recent study [22] showing that router technology has been clearly producing higher degree-capacity combinations at core routers in comparison to the edge routers.

Traffic Demand Model

A crucial piece in modeling an ISP network is the workload model, i.e., a traffic matrix. In addition to being realistic in size, each traffic flow in the network model must reflect the traffic from edge router to another edge router. Thus, there are two important steps in constructing a reasonable traffic matrix. First, we identify the edge routers from the Rocketfuel topologies by picking the routers with smaller degree or longer distance from the center of the topology. To do so, for each of the Rocketfuel topologies, we identified *Degree Threshold* and *BFS Distance Threshold* values so that the number of edge routers corresponds to 75-80% of the nodes in the topology. Second, we use gravity models [27, 50] to construct a feasible traffic matrix composed of edge-to-edge (G2G) flows. The essence of the gravity model is that the traffic between two routers should be proportional to the product of the populations of the two cities where the routers are located. We used CIESIN [2] dataset to calculate the city populations. We construct an initial traffic matrix based on the gravity model using populations of the cities, and then adjust the BFS-bases link capacity estimations (see Section 5.1.1) so that traffic load on individual links are feasible. This method of generating traffic matrices based on gravity models yields a power-law behavior in the flow rates as was studied earlier [25, 27]. We assume that this final traffic matrix reflects the state of the network in a steady state condition. During the simulation, we base our work on this initial condition and analyze the transitions from this initial state of the network.

5.1.2 Inter-domain Topology

For creating inter-domain topology, we use BRITE topology generator [26]. We select AS-level topology creation according to Albert-Barabasi method [4] which models scale free characteristics of Internet. Our topology size is 15 for the following set of simulations.

In AS-level topology produced by BRITE, provider domains represented as single hops. Neighboring relationships between these single hops are given as output. For our simulations, we simply replace these single hops with randomly chosen realistic Rocketful topologies and have a router-level topology consisting of 15 ISPs with realistic intra-domain models. To achieve this, we simply revise SSFNet integration modules of BRITE so as to feed Rocketfuel intra-domain topology maps and to embed them into AS-level topology generated by BRITE by following procedure:

First assign a random Rocketfuel intra-domain topology for each domain in inter-domain topology.

1. Add all ISPs into set S
2. Continue while set S is not empty
3. Select the most connected ISP X in set S
4. Select the most connected edge router on ISP X (router resides in city A)
5. Select the most connected neighbor ISP Y of ISP X which ISP X still does not have a established peering point.
6. Select the edge router of ISP Y which resides in the city A or the closest city to city A

7. Establish a peering point between these edge routers.

8. If all peering points are assigned for ISP X, remove ISP X from set S

Above heuristic will continue by establishing peering relationships between topologies at a common PoP point within the same (or closest) city if possible.

5.2 Evaluation

Our evaluation part consists of two parts. In the first part, we examine robustness and efficacy of contract links on realistic network topologies. These characteristics are crucial since Contract Switching relies on contract link definitions as building blocks. Second part mostly focuses on revealing the Contract Routing behavior on high load and moderate cases so as to underline how Contract Routing performs under these conditions and how system parameters affect this performance. Second part also includes several cost figures for contract-routing protocol overhead assessment.

5.2.1 Contract Link Evaluation

We analyze contract link efficiency for both economic and network performance perspectives. For economic analysis, we first define one specific contracting mechanism which is called "Bailout Forward Contracting (BFC)" as our contract link definition. First we need to give definitions of forward and bailout forward terms:

A Forward Contract

A forward contract is an obligation for delivering a (well-defined) commodity (or service) at a future time at a predetermined price - known as the 'Forward Price'.

Other specifications of the contract are Quality Specification and Duration (start time - T_i, and end time - T_e, for the delivery of a timed service).

A Bailout Forward Contract (BFC)

In the case of a capacitated resource underlying a forward contract, restrictions may be necessary on what can be guaranteed for delivery in future. A key factor that defines the capacity of the resource is used to define the restriction. A bailout clause added to the forward contract releases the provider from the obligation of delivering the service if the bailout clause is activated, i.e. the key factor defining the capacity rises to a level making delivery of the service infeasible. A set up is essential for the two contracting parties to transparently observe the activation of the bailout clause in order for the commoditization of the forward contract and elimination of moral hazard issues. The forward price associated with a bailout forward contract takes into account the fact that in certain scenarios the contract will cease to be obligatory.

Risk Evaluation

A bailout forward contract on a capacitated resource enables risk segmentation and management of future uncertainties in demand and supply of the resource. Contracts are written on future excess capacity at a certain price, the forward price, thus guaranteeing utilization of this capacity; however if the capacity is unavailable at the future time, the bailout clause allows a bailout. Therefore, it hedges the precise segment of risk. The price of the bailout forward reflects this.

For economic risk evaluation, first we select a subset of edge-to-edge traffic flows for further analysis. Our selection will favor traffic flows following long-haul edge-to-edge paths where egress and ingress routers are separated at least by a distance

threshold which is a fraction of the average distance between cities of a given topology. Then, we fail each link which has a non-zero traffic load and whose failure does not lead disconnected subnetworks within topology. Number of selected flows and number of all flows can be found in Table 5.2 and Table 5.3. After examining edge-to-edge bandwidth capacity after each single link failure, we calculate average bandwidth capacity and standard deviation values for each edge-to-edge path. We plug these statistical parameters into our mathematical model to produce future projections on edge-to-edge available bandwidth capacity in according to our stochastic Wiener process. Similarly, another projection model leverages these statistics to produce future projections on traffic demand.

Another important factor is that edge-to-edge contract links may share physical resources along the way. Through load fluctuations or topology changes upon component failures, contract links may affect performance of other contract links adversely. So, we have to model these interactions between traffic flows to resolve correlation among them.

Intensity of Overlap

To evaluate the risk involved in advertising a particular G2G contract, knowledge of the interactions among crossing flows within the underlying network is crucial. As mentioned in the previous subsection, we develop our multiple G2G BFC terms based on the assumption that an intensity of overlap, ρ^{ij}, abstractly models the correlation between flows i and j. High correlation means that flows i and j are tightly coupled and share more of the network resources on their paths. In other words, an increase in flow i's traffic will adversely affect the available G2G capacity for flow j and vice versa.

We construct the correlation information among the G2G contracts as a square matrix of overlapping links. Each entry of ρ^{ij} reflects the overall effect of flow i on flow j which is the result of the contention that takes place on common links that two flows overlap on their E2E paths. Contention becomes severe if a race condition exists between flows for limited bandwidth on a link. We model this contention as being dominated by the *severity of contention* at the bottleneck link on the G2G path. Thus, we pick the severity of contention on the most utilized common link as the indicator of the correlation between the two overlapping flows.

In our calculation, we also reflect the utilization level of bottleneck link as an indicator of severity of race condition among the flows.

Also, we consider the *asymmetric* characteristic of the overlaps arising due to the amount of individual traffic which are not necessarily equal. So, the effect of flow i on flow j, is not necessarily equal to the effect of flow j on flow i. In that sense, the effect of flow i on j is proportional to the ratio of traffic that flow i generates to the overall traffic generated by this flow pair.

Thus, we model the correlation between flows i and j as:

$$\rho^{ij} = U_{link} \times \left(\frac{\tau_i}{\tau_i + \tau_j} \right)$$

where τ_k is the portion of bandwidth that flow k can have according to max-min fair share among all flows passing through the common bottleneck link, and U_{link} is the utilization of the bottleneck link. To calculate τ_k for flow k, first we calculate bandwidth distribution over every single link using E2E demand for flow k (i.e., μ_t^k) and the available link capacities (i.e., $C_{N\times N}$). More specifically, on the common bottleneck link, we distribute the available capacity to all passing flows according to

max-min fair share. Then, we distributed the excess capacity evenly across all the flows until no excess capacity is left on the link. This strategy makes τ_k being the minimum capacity allocated to flow k over all links it passes through.

Determining BFC Terms

Once we have above mentioned models for traffic demand, bandwidth supply and correlation among traffic flows, we build our pricing mechanism so as to calculate risk-neutral prices for pricing our bailout forward contracts. Beside risk-neutral prices, we also determine bailout clauses for BFCs. We run our simulations 1000 times and for each edge-to-edge path, we pick the 15th percentile of available bandwidth capacity of edge-to-edge path as the bailout clause for contract links defined on that edge-to-edge path. According to that, if edge-to-edge bandwidth capacity goes below that level, then provider bails out the contract.

Network Performance

Table 5.5 and 5.6 give the fraction of bailout contracts after topology changes upon link failure scenarios for 6 Rocketfuel topologies. As seen in results, for all topologies contract links are robust to topology changes over 85% of the time. Robustness values are slightly higher for well-engineered topologies (e.g. Sprint and Tiscali) than the hub-and-spoke topologies (e.g. Exodus and Ebone). It is important to note that route restoration in our model is limited to what OSPF is capable of. Considering advanced restoration, fast rerouting, tunneling and traffic engineered back up path technologies which can achieve network stability under 100 miliseconds, contract link robustness and performance under OSPF restoration case can be considered as a lower bound. Having said that, even under OSPF restoration case, contract links are robust and

robust enough to be profitable and efficient to be used in multi-hop contract path establishments.

Economic Perspective

At this stage of our research, we focus specifically on network performance of contract-routing protocol. However, we want to share some significant simulation results for our economic model in short. First of all, we run our pricing and demand-supply projection models on Exodus topology only. We specifically focus on price segmentation and economic feasibility of contract link definitions.

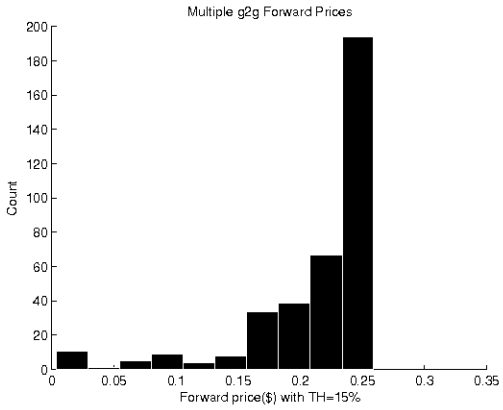


Figure 5.1: Price Segmentation

In Figure 5.1, we plot the forward prices of 372 contract links of Exodus topology in a histogram. As the histogram suggests, although there is variability in the forward prices across the set of paths, many of the paths pick a forward price in a similar range, in this case approximately around 0.25. This suggests that a distinct forward price for each of the thousands of G2G paths in a topology may be an overkill, and hence, directs us to a much desired simplicity in the forward pricing structure.

Figure 5.2 summarizes the comparison of loss in revenue and fraction of paths

Case	Expected Total Revenue	Mean Bailout Fraction
Artificial No Bailout or Failure Case	**95.7464**	0
Base Case Bailout Scenario	80.43655	0.16403
Bailouts in Failure Mode 1	78.98833	0.16505355
Bailouts in Failure Mode 2	81.34074	0.163980954
Bailouts in Failure Mode 3	80.98213	0.16676308

Figure 5.2: Revenue Analysis

bailing out in the four scenarios - the base case and the three failure modes. For these failure modes, we select the scenarios where the most loaded link of the given topology is failed. There is only a small increase in the fraction of paths bailing out in the failure modes, as well as only a small reduction in revenue from the base case. This is supporting evidence for the robustness of the BFC framework.

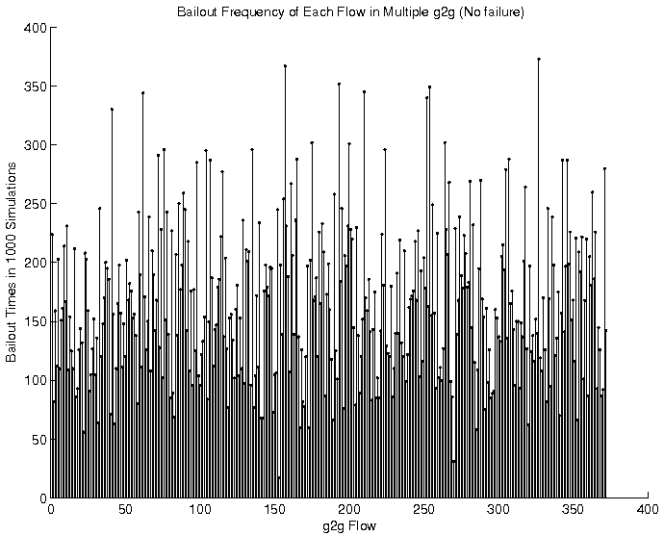


Figure 5.3: Number of times Contract Links Bailout

Bailout characteristics are the next important feature to study to evaluate the BFC framework. We plot the fraction of 372 G2G paths bailing out in 1000 runs of simulation in a histogram in Figure 5.4. The mean fraction of G2G paths bailing out from this histogram is 0.16403, or 16.4%. To highlight which specific paths bail out

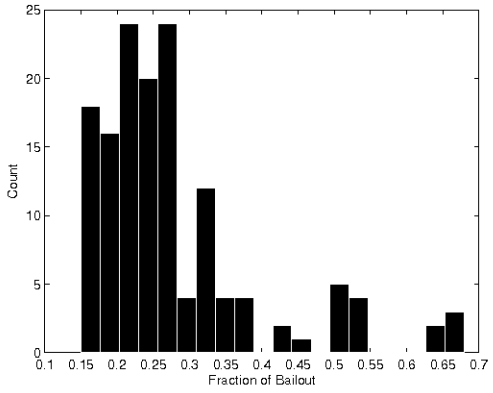


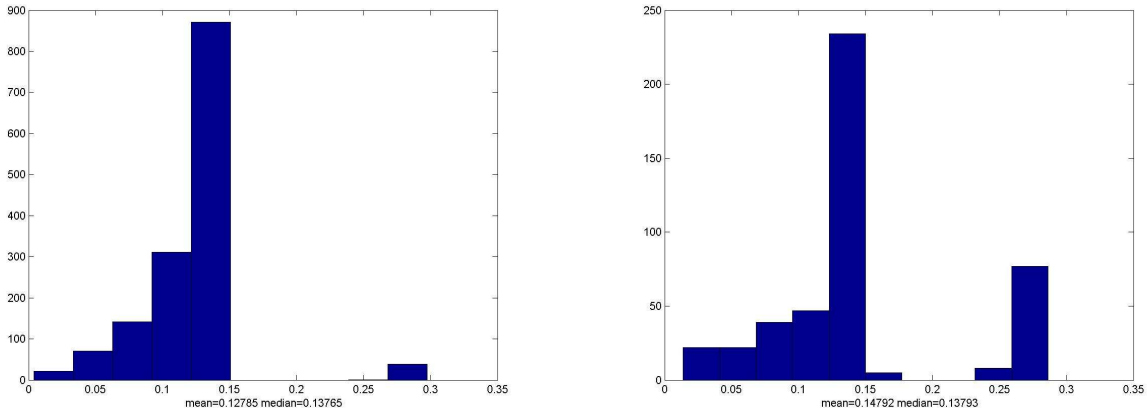
Figure 5.4: Bailout Histogram for Exodus topology

in these simulation runs, we also plot the number of times each link bails out in the 1000 runs of simulation in Figure 5.3. There are a few paths that clearly stand out in bailing out most frequently, marking the 'skyline', while most of the paths cluster in the bottom. Another important measure of performance is how much revenue is lost when the BFC on a G2G path bails out.

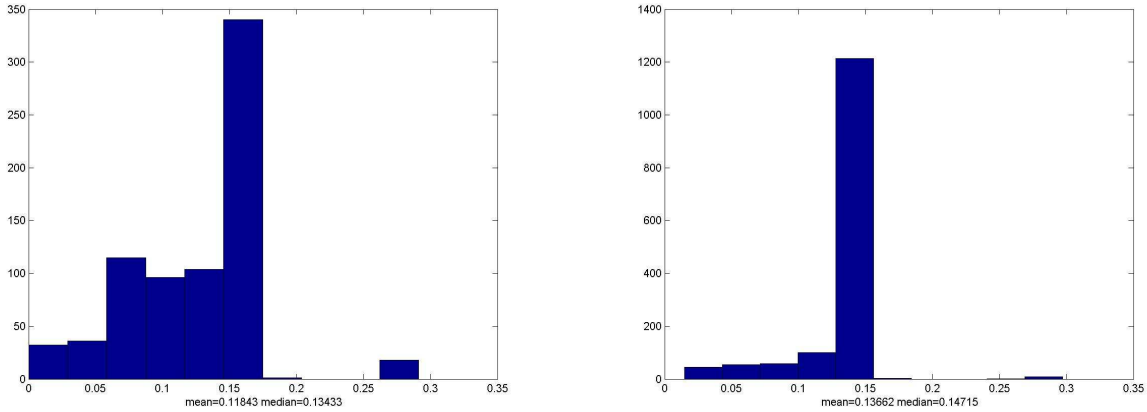
Table 5.3: Rocketfuel-based router-level ISP topologies.

ISP	# of Failed Links	# of Selected G2G Flows
Abovenet	290	454
Ebone	170	390
Exodus	160	372
Sprintlink	494	1456
Telstra	134	742
Tiscali	333	1484

In this subsection we investigate how robust contract link definition is under intra-domain topology simulations of 6 Rocketfuel topologies and we see that contract link definition is a robust and economically feasible tool to deploy. In the following subsections, we will investigate contract-routing behavior in an inter-domain topology level whose setup is described in Subsection 5.1.2.



(a) Sprint (b) Abovenet

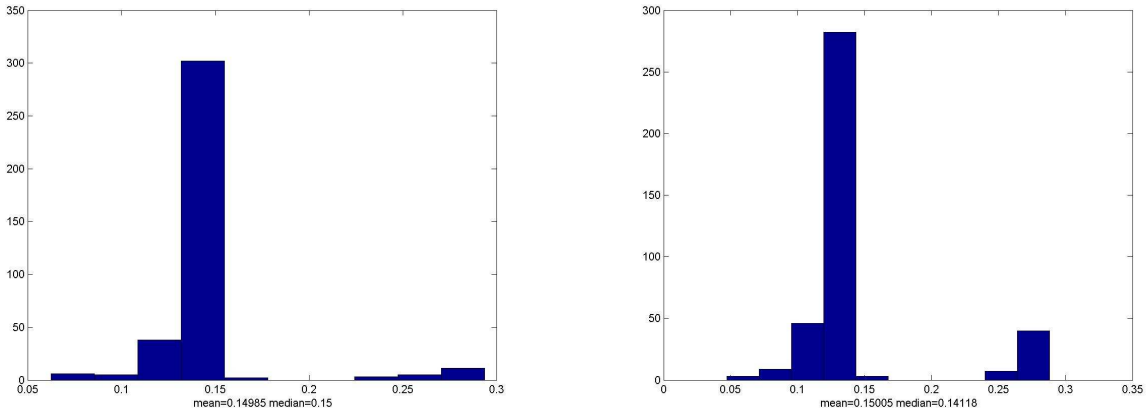


(c) Telstra (d) Tiscali

Figure 5.5: Contract Link Robustness (# of simulations vs ratio of bailing out contracts)

5.2.2 QoS vs Reachability Tradeoff

Contract term is a significant parameter on pricing calculations and risk segmentation. Degree of future unpredictability and risks of commitment made by contract link vary with the duration of contract term. Beside these theoretical well-known roles, another interesting question arises is that: How does the contract terms affect the contract-routing performance? To evaluate contract term effects on system performance, we



(e) Exodus (f) Ebone

Figure 5.6: Contract Link Robustness (# of simulations vs ratio of bailing out contracts)

decide to run our simulations for high load cases where traffic demand is well over network capacity. The reason for this decision is two-fold, first system behavior is magnified under extreme cases so that it revealed itself easily and the second one is that we predict that in such a Contract Switched Market both providers and customers behave opportunistically so it can be expected that there may be times where providers prefer operating at low supply (or high utilization) level cases.

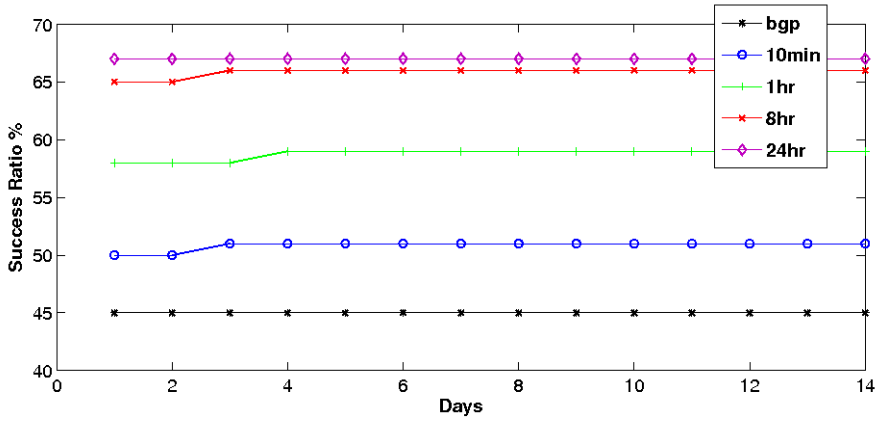


Figure 5.7: QoS vs Reachability Dilemma under High Load: QoS Performance

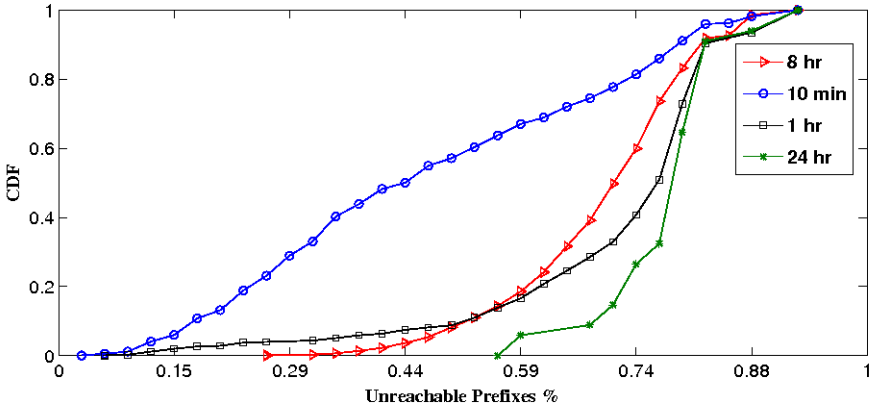


Figure 5.8: QoS vs Reachability Dilemma under High Load: Unreachable Prefixes

As Figure 5.7 depicts, for all duration of contract terms, contract-routing over-performs BGP in terms of bandwidth capacity of paths which traffic flows take (even at the case of extreme 10 minutes contract term). Actually these results are not surprising since BGP is constrained within shortest path criterion whereas contract-routing can explore relatively low utilized paths and can make load balancing over multiple paths.

Another important result is that Figure 5.8 in combined with Figure 5.7 shows that as contract term gets longer, contract-routing becomes more able to sustain higher QoS levels. But as contract term gets longer, edge-to-edge capacity reserved by established contract paths increase and as a result of that it gets harder for a contract router to establish new contract paths due to increasing number of unreachable destinations.

5.2.3 Price Convergence

Another effect of contract term is that as contract term gets longer, it takes more time for the market to reach price stability. According to Figure 5.9, price instability

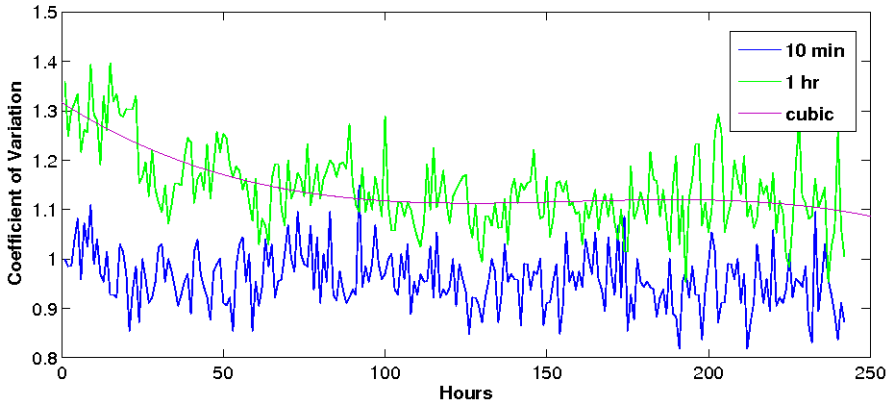


Figure 5.9: Price Stability

represented by coefficient of variation of contract link prices are higher for 1 hour long contract term than 10 minutes contract term. It can be said that shorter contract terms allow more interactions so that market quickly explores stable resource and price arrangements.

5.2.4 Protocol Overhead

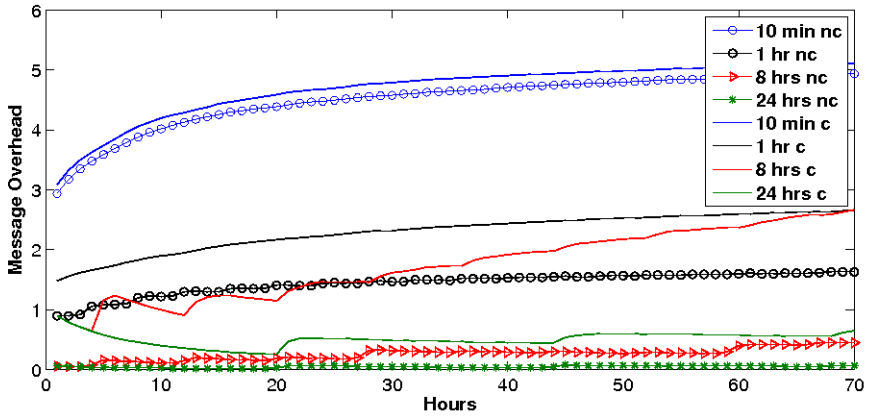


Figure 5.10: Contract Routing Message Overhead (c for high load and nc for moderate load)

One of the important characteristics of a routing protocol is messaging over-

head. To compare messaging overhead of contract routing, we plot the ratio of number of link state update messages of contract routing to the number of BGP update messages. For both high load and moderate load cases as seen in Figure 5.10, as contract term gets longer messaging overhead decreases. For extreme 10 minutes long contracts, messaging overhead could be as high as 5 times of number of BGP update messages. For all other cases, messaging overhead is well under 2.5 times of what BGP incurs. For moderate load, they are only a fraction of the number of BGP update messages.

Contract Routing deviates from shortest path routing so as to discover more adventageous end-to-end paths in terms of better quality (or less cost). To investigate this deviation quantitatively, we simulate both moderate and high load cases and compare the length of BGP routed shortest paths with length of established contract paths.

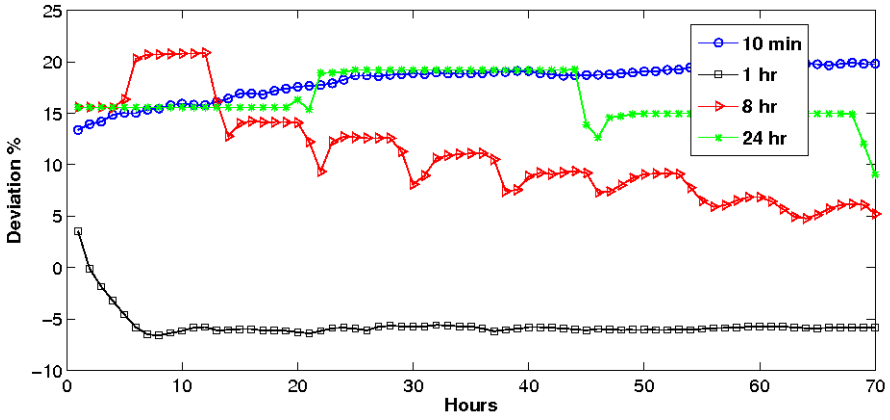


Figure 5.11: Path Length Comparison: High Load

As seen in Figure 5.11, contract-routing may deviate up to 20% from shortest path. As contract term gets longer, deviation is increased. Here, 10 minutes contract term can be described as outlier since all 24 hours, 8 hours and 1 hour contract

term behaviors are in parallel. Since shorter contract terms lead quick stabilization of the system, with shorter contract term contract-routing becomes more capable of discovering shorter paths among contract paths with similar quality in terms of bandwidth capacity and with similar cost. This behavior reveals itself in decreasing path length characteristics as well as price stabilization.

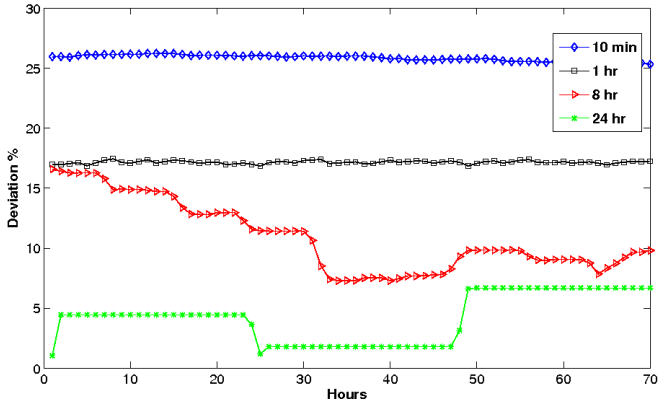


Figure 5.12: Path Length Comparison: Moderate Load

For moderate load cases as seen in Figure 5.12, deviation pattern is reversed. So, as contract term gets longer, deviation is increased again up to 20%. This is due to our greedy proactive contracting function which leads over-contracting in favor of uninterrupted QoS level for end-to-end services.

Chapter 6

Discussion

In a market where provider compensation is not defined in proportional to provider investments and contribution, service providers will not be willing to make new investments on innovative QoS technologies. Current Internet architecture with point-to-anywhere service definition and static contracting mechanisms could not provide flexible provider compensation models to overcome the above problem. Contract-Switching approach introduces dynamic contracting mechanisms so as to create a market where both customers and providers benefit from their increased expression power of choices and preferences.

In this work, our greatest contribution is that we showed that end-to-end guaranteed QoS services can be achieved through contract link abstractions which are built on today's popular protocols and current Internet technologies. Our simulation results show that contract links are robust even in case of drastic topology changes and major network element failures. Moreover, our economic analysis on Bailout Forward Contracts reveals that routing over contract links is economically feasible. Furthermore, we also implement and evaluate performance of Link State Contract

Routing (LSCR) protocol for macro time-level contract terms (e.g. hours and up to weeks). We showed that routing over contract links with LSCR can be achieved with reasonable messaging overhead and with limited deviation from shortest path routing efficiency.

For our future work, we first plan to implement Path Vector Contract Routing (PVCR) protocol for micro time-level contract terms (e.g. as short as minutes and up to hours). We first have to show that satisfactory route availability and end-to-end quality of service can be achieved in such a dynamic framework with on-demand and on-line manner within micro-level durations. Then, our next step is to show that through interplay of concurrently running PVCR and LSCR protocols, realistic user demand for guaranteed end-to-end QoS services could be met in Contract-Switching architecture feasibly. One additional future task is to demonstrate that LSCR protocol converges within expected time span (tens-of-minutes) in a larger set of realistic inter-domain topology.

Our ultimate goal is to examine Contract-Switched Market where dynamic contracting schemes are well adapted and both customers and providers interact in a well structured market model. Such a market will be definitely much different than today's Internet market. Our research differs from other future Internet architecture proposals with the introduction and involvement of well-described economic tools and incentivizing mechanisms for structural change. So, we think that our approach will contribute the discussions on future Internet architecture and economic models supporting such a market.

Bibliography

[1] Arbinet-thexchange, Inc. `www.arbinet.com`.

[2] The Center for International Earth Science Information Network (CIESIN). `http://www.ciesin.columbia.edu`.

[3] The internet singularity, delayed: Why limits in internet capacity will stifle innovation on the web. Nemertes Research, 2007.

[4] R. Albert and A.-L. Barabasi. Topology of evolving networks: local events and universality. *Physical Review Letters*, 85:5234, 2000.

[5] A. Barbir, R. Penno, R. Chen, M. Hofmann, and H. Orman. An architecture for open pluggable edge services (opes). RFC 3835, August 2004.

[6] R. Y. . K. D. Bates T., Chandra R. Multiprotocol extensions for bgp-4. IETF RFC 4760, January 2007.

[7] R. Braden and et al. Resource Reservation Protocol (RSVP) - V1 functional Spec. *IETF Internet RFC 2205*, Sep 1997.

[8] K. Calvert, M. Doar, and E. W. Zegura. Modeling internet topology. *IEEE Communications Magazine*, 35(6):160–163, June 1997.

[9] I. Castineyra, N. Chiappa, and M. Steenstrup. The nimrod routing architecture. *IETF RFC 1992*, August 1996.

[10] D. D. Clark, J. Wroclawski, K. R. Sollins, and R. Braden. Tussle in cyberspace: defining tomorrow's internet. In *SIGCOMM '02: Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 347–356, New York, NY, USA, 2002. ACM.

[11] R. C. E. Rosen, A. Viswanathan. Multiprotocol label switching architecture. IETF RFC 3031, January 2001.

[12] N. Feamster, H. Balakrishnan, J. Rexford, A. Shaikh, and J. van der Merwe. The case for separating routing from routers. In *FDNA '04: Proceedings of the ACM SIGCOMM workshop on Future directions in network architecture*, pages 5–12, New York, NY, USA, 2004. ACM.

[13] J. He and J. Rexford. Toward internet-wide multipath routing. *Network, IEEE*, 22(2):16–21, March-April 2008.

[14] J. D. Houle, K. K. Ramakrishnan, R. Sadhvani, M. Yuksel, and S. Kalyanaraman. The evolving internet - traffic, engineering, and roles. In *Proc. of Research Conference on Communication, Information and Internet Policy (TPRC)*, Arlington, VA, September 2007.

[15] E. Inc. http://www.equinix.com, 2004.

[16] W. Jiang, R. Zhang-Shen, J. Rexford, and M. Chiang. Cooperative content distribution and traffic engineering. In *NetEcon '08: Proceedings of the 3rd international workshop on Economics of networked systems*, pages 7–12, New York, NY, USA, 2008. ACM.

[17] K. Kar, M. Kodialam, and T. V. Lakshman. Minimum interference routing of bandwidth guaranteed tunnels with mpls traffic engineering applications. *IEEE Journal on Selected Areas in Communications*, 18:2566–2579, 2000.

[18] J. Kempf and R. Austein. The rise of the middle and the future of end-to-end: Reflections on the evolution of the internet architecture. *IETF Internet RFC 3724*, March 2004.

[19] V. Krishnamurthy, M. Faloutsos, M. Chrobak, J.-H. Cui, L. Lao, and A. G. Percus. Sampling large internet topologies for simulation purposes. *Comput. Netw.*, 51(15):4284–4302, 2007.

[20] K. K. Lakshminarayanan, I. Stoica, S. Shenker, and J. Rexford. Routing as a service. Technical Report UCB/EECS-2006-19, EECS Department, University of California, Berkeley, Feb 2006.

[21] P. Laskowski, B. Johnson, and J. Chuang. User-directed routing: from theory, towards practice. In J. Feigenbaum and Y. R. Yang, editors, *NetEcon*, pages 1–6. ACM, 2008.

[22] L. Li, D. Alderson, W. Willinger, and J. Doyle. A first principles approach to understanding the internet's router-level topology. In *Proc. of ACM SIGCOMM*, 2004.

[23] W. Liu, H. T. Karaoglu, A. Gupta, M. Yuksel, and K. Kar. Edge-to-edge bailout forward contracts for single-domain internet services. In *Proceedings of IEEE International Workshop on Quality of Service (IWQoS)*, Enschede, Netherlands, June 2008.

[24] D. Magoni and J.-J. Pansiot. Internet topology modeler based on map sampling. In *ISCC '02: Proceedings of the Seventh International Symposium on Computers and Communications (ISCC'02)*, page 1021, Washington, DC, USA, 2002. IEEE Computer Society.

[25] R. Mahajan, D. Wetherall, and T. Anderson. Negotiation-based routing between neighboring isps. In *Proc. of USENIX NSDI*, 2005.

[26] A. Medina, A. Lakhina, I. Matta, and J. Byers. Brite: An approach to universal topology generation. In *MASCOTS '01: Proceedings of the Ninth International Symposium in Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, page 346, Washington, DC, USA, 2001. IEEE Computer Society.

[27] A. Medina, N. Taft, K. Salamatian, S. Bhattacharyya, and C. Diot. Traffic matrix estimation: Existing techniques and new directions. In *Proc. of ACM SIGCOMM*, 2002.

[28] J. Moy. OSPF version 2. *IETF RFC 2328*, April 1998.

[29] V. Paxson. End-to-end routing behavior in the internet. *IEEE/ACM Trans. Netw.*, 5(5):601–615, 1997.

[30] V. Paxson and S. Floyd. Why we don't know how to simulate the internet. In *Proceedings of the 1997 Winter Simulation Conference*. SCS, December 1997.

[31] C. Perkins. Ip encapsulation within ip. IETF RFC 2003, October 1996.

[32] Y. Rekhter and T. Li. A border gateway protocol 4 BGP-4. *IETF RFC 1771*, March 1995.

[33] J. Rexford, J. Wang, Z. Xiao, and Y. Zhang. Bgp routing stability of popular destinations. In *IMW '02: Proc. of ACM SIGCOMM*, pages 197–202, 2002.

[34] J. H. Saltzer, D. P. Reed, and D. D. Clark. End-to-end arguments in system design. *ACM Trans. Comput. Syst.*, 2(4):277–288, 1984.

[35] S. Savage, A. Collins, E. Hoffman, J. Snell, and T. Anderson. The end-to-end effects of internet path selection. In *In Proc. of ACM Sigcomm*, pages 289–299, 1999.

[36] Y. Shavitt and E. Shir. Dimes: Let the internet measure itself, 2005.

[37] Y. Shavitt and Y. Singer. Trading potatoes in distributed multi-tier routing systems. In *NetEcon '08: Proceedings of the 3rd international workshop on Economics of networked systems*, pages 67–72, New York, NY, USA, 2008. ACM.

[38] N. Spring, R. Mahajan, and D. Wetherall. Measuring isp topologies with rocketfuel. In *Proceedings of Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, 2002.

[39] As external lsa support for SSFNet. `http://cnl.cse.unr.edu/?c=resources`.

[40] SSFNet – scalable simulation framework. `http://www.ssfnet.org`.

[41] L. Subramanian, M. Caesar, C. T. Ee, M. Handley, M. Mao, S. Shenker, and I. Stoica. Hlp: a next generation inter-domain routing protocol. In *SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 13–24, New York, NY, USA, 2005. ACM.

[42] R. Teixeira, S. Agarwal, and J. Rexford. Bgp routing changes: Merging views from two isps. *ACM SIGCOMM Computer Communication Review (CCR)*, October 2005.

[43] R. Teixeira, K. Marzullo, S. Savage, and G. M. Voelker. Characterizing and measuring path diversity of internet topologies. In *SIGMETRICS '03: Proc. of the 2003 ACM SIGMETRICS*, pages 304–305, New York, NY, USA, 2003. ACM.

[44] J. Winick and S. Jamin. Inet-3.0: Internet topology generator.

[45] W. Xu and J. Rexford. Miro: multi-path interdomain routing. *SIGCOMM Comput. Commun. Rev.*, 36(4):171–182, 2006.

[46] X. Yang, D. Clark, and A. Berger. Nira: A new inter-domain routing architecture. *IEEE/ACM Transactions on Networking (ToN)*, 15(4):775–788, August 2007.

[47] C. S. Yoo. Public En Banc Hearing on Broadband Network Management Practices Before the Federal Communications Commissions, 2008. `http://www.law.upenn.edu/cf/faculty/csyoo/`.

[48] M. Yuksel, A. Gupta, and S. Kalyanaraman. Contract-switching paradigm for internet value flows and risk management. In *Proceedings of IEEE Global Internet Symposium*, 2008.

[49] M. Yuksel and S. Kalyanaraman. Distributed dynamic capacity contracting: A congestion pricing framework for Diff-Serv. *Proceedings of IFIP/IEEE International Conference on Management of Multimedia Networks and Services (MMNS)*, Oct 2002.

[50] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg. Fast accurate computation of largescale ip traffic matrices from link loads. In *Proc. of ACM SIGMETRICS*, 2003.

[51] D. Zhu, M. Gritter, and D. R. Cheriton. Feedback based routing. *SIGCOMM Comput. Commun. Rev.*, 33(1):71–76, 2003.

[52] Y. Zhu, R. Zhang-Shen, S. Rangarajan, and J. Rexford. Cabernet: connectivity architecture for better network services. In *CONEXT '08: Proceedings of the 2008 ACM CoNEXT Conference*, pages 1–6, New York, NY, USA, 2008. ACM.