A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Computer Science and Engineering

by

Hasan T Karaoglu

Dr. Murat Yuksel
Dissertation Advisor

August, 2012
Copyright
by
Hasan Tarik Karaoglu
© 2012
THE GRADUATE SCHOOL

We recommend that the dissertation prepared under our supervision by

HASAN TARIK KARAOGLU

entitled

Contract Routing Architecture

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

DOCTOR OF PHILOSOPHY

Murat Yuksel, Ph.D., Advisor

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

Frederick C. Harris, Jr., Ph.D., Committee Member

Monica Nicolescu, Ph.D., Committee Member

Thomas Quint, Ph.D., Graduate School Representative

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

August, 2012
Abstract

The current Internet architecture has been principally designed for overcoming the interoperability issues between previously disconnected network islands. Since the focus was to ensure openness and connectivity, the architecture maintained simplistic core functionalities, i.e., the set of the capabilities that have to be compatible among the Internet stakeholders. Routing, which is the negotiation of end-to-end data paths, has been successfully handled so far, since the only expectation from the Internet was to provide the basic end-to-end connectivity. Today, however, customers are not only looking for a connectivity that just works but also supports a plethora of applications that require low-latency, high-bandwidth, and secure connections to the rest of the world. Such value-added services cannot be realized effectively mainly because the Internet core is economically rigid and only allows bilateral, single-metric negotiation of paths in a coarse-grained manner. The revenue sharing model built on this rigid economics is unhealthy and incapable of addressing the long-term sustainability and innovation at all parts (i.e., core, edge, and end) of the Internet. To overcome these limitations, we propose Contract Routing Architecture that enables users and Internet Service Providers (ISPs) i) to express their perceived economic value of offered services, and ii) to negotiate end-to-end quality of paths with other stakeholders in a multi-lateral manner considering multiple performance metrics. In this thesis, we describe how multi-metric, multi-hop path negotiations can be achieved in a scalable manner while making economics an inherent part of routing process itself so as to amend broken revenue model of the current Internet.
To my mom and dad
Acknowledgments

First of all, I would like to thank my family for their unconditional support through good and bad times. Then, I would like to thank my advisor, Dr. Murat Yuksel, for his trust on me even at the most difficult times. Also, many thanks to committee members Dr. Mehmet H. Gunes, Dr. Monica Nicolescu, Dr. Frederick Harris and Dr. Thomas Quint for their valuable comments. Finally, I would like to thank Beyzade Sahin as his idealism and dedication to teaching inspired me in my early life.

Hasan Tarik Karaoglu

University of Nevada, Reno

August 2012
# Contents

Abstract

Acknowledgments

List of Figures

Chapter 1 Introduction

1.1 Internet Architecture and Current Issues

1.2 Organization of the Dissertation

Chapter 2 Related Work

2.1 Improvement Proposals

2.2 Clean Slate Approaches

Chapter 3 Contract Routing Architecture

3.1 Contract Definition

3.1.1 Contract Link Components

3.1.2 Contract Link Types

3.2 Architecture and Modules

3.2.1 Design Rationale
3.2.2 Modular Design ........................................ 22
3.2.3 Network Elements ..................................... 23
3.2.4 Proposed Design ....................................... 24
3.3 Link State Contract Routing ............................... 26
3.4 Path Vector Contract Routing ............................. 28
3.4.1 Path Explosion in PVCR ............................... 31
3.4.2 Routing Performance and Scalability Analysis .......................... 36
3.4.3 Traffic Engineering with PVCR ........................ 39
3.4.4 PVCR Performance on PoP-level Topology ............ 48

Chapter 4 Benefit Analysis ................................. 54
4.1 Technical Benefit Analysis from Network Perspective .......... 54
4.1.1 Effectiveness of Current Routing Policies .................... 56
4.1.2 k-step Local Improvement Policy .......................... 57
4.1.3 Policy Performance Evaluation ............................ 59
4.1.4 Implications for Clean-Slate Internet Architecture Proposals .... 66
4.1.5 Summary and Further Issues .............................. 69
4.2 Economic Benefit Analysis from Risk Management Perspective ...... 69
4.2.1 Bailout Forward Contracts (BFCs) .......................... 69
4.2.2 Economic Benefit Analysis of BFCs ........................ 74

Chapter 5 Cloud-Assisted Routing ......................... 85
5.1 Cloud-Assisted Routing .................................. 87
5.1.1 CAR Architectural View and Principles ................. 87
5.1.2 CAR Motivations ....................................... 91
5.1.3 CAR Challenges and Tradeoffs .......................... 92
5.2 Proposed Tasks ................................................................. 93
  5.2.1 CAR I: Peak CPU Utilization for BGP Peering ............... 93
  5.2.2 CAR II: BGP Forwarding Table Size ............................. 95
  5.3 Previous Research .......................................................... 96

Chapter 6 Conclusions and Future Work ................................. 99
  6.1 Conclusions ................................................................. 99
  6.2 Future Work ............................................................... 100

Bibliography .......................................................... 101
## List of Figures

2.1 Path Trading Approach [71] ........................................... 8
2.2 Multipath Routing [109] ........................................... 9
2.3 Taken paths are embedded in hierarchical addressing by NIRA [110] . 11
2.4 HLP [93] ................................................................. 13

3.1 Packet Switching and Contract Switching . ................................. 15
3.2 Contract Link Abstraction ............................................... 18
3.3 Contract Link Abstraction ............................................... 19
3.4 Contract Routing Framework ............................................. 23
3.5 Network Elements ....................................................... 24
3.6 Scenario for LSCR ....................................................... 25
3.7 Path-vector contract routing: (a) Provider initiates. (b) User initiates 29
3.8 Path-Vector Contract Routing Path Negotiation [57] .................... 32
3.9 Path Exploration Success (95th percentile CI) ......................... 37
3.10 Path Diversity: Number of Path Explored (95th percentile CI) ....... 37
3.11 Path Stretch (95th percentile CI) ................................... 38
3.12 Total Number of Unique Path Request Messages (95th percentile CI) 39
3.13 Total Path Request Message Traffic (95th percentile CI) ............ 40
3.14 Congestion-Aware Price Function ........................................ 43
3.15 Network Price Map on a 10x10 Grid Topology .......................... 44
3.16 Network Price Map on a 10x10 Grid Topology .......................... 45
3.17 Network Price Map on a 10x10 Grid Topology .......................... 47
3.18 Network Price Map on a PoP-level iPlane topology at lower utilization levels ................................................................. 52
3.19 Network Price Map on a PoP-level iPlane topology at higher utilization levels ................................................................. 53
4.1 Spatial granularity shortcomings of current policies. ................. 57
4.2 Average routing performance improvements of k-step local improve- ment routing over hot-potato routing (hops as PoPs) ..................... 60
4.3 Normalized average routing performance improvements of k-step local improvement routing over hot-potato routing (hops as PoPs) ............. 60
4.4 Latency improvements analyzed side-by-side with the number of sample Internet routes with various path length in # of PoPs ............... 62
4.5 Loss rate improvements analyzed side-by-side with the number of sample Internet routes with various path length in # of PoPs .......... 62
4.6 Latency improvements of k-step local improvement routing over hot- potato routing for various path lengths in # of PoPs .................... 63
4.7 Normalized latency improvements of k-step local improvement routing over hot-potato routing for various path lengths in # of PoPs ........ 63
4.8 Loss rate improvements of k-step local improvement routing over hot- potato routing for various path lengths in # of PoPs .................... 64
4.9 Normalized loss rate improvements of k-step local improvement routing over hot-potato routing for various path lengths in # of PoPs ........ 64

4.10 Number of ISPs on the Global Optimum Path According to Latency and Loss Rate Improvement Policy ......................... 65

4.11 ABOVENET: Top panel: 95% Confidence Intervals for Mean Revenue for each link of Node 1 and Node 5 in BC2. Bottom panel: Mean Revenue level for each group of links of Node 1 and Node 5 in Reduced BC2. ......................... 80

4.12 ABOVENET: Histograms of 7 Day Total Revenue from Baseline Case 1, Baseline Case 2, and Reduced Baseline Case 2 for Nodes 1 and 5. . 80

4.13 Left panel: ABOVENET: 95% Confidence Intervals for Mean Revenue in BFC, and Mean Revenue level after grouping in Reduced BFC (Node 1). Right panel: EXODUS: 95% Confidence Intervals for Mean Revenue in BFC, and Mean Revenue level after grouping in Reduced BFC (Node 2). ......................... 81

4.14 ABOVENET: Histograms of 7 Day Total Revenue for Baseline Case 1, Reduced Baseline Case 2, and Reduced BFC with demand conversion at 40% (Node 1) ......................... 82

4.15 ABOVENET: Histograms for 7 Day Total Revenue from Reduced BC2 (top-left) and Reduced BFC with different demand conversion rate (CR = 20% (top-right), 30% (bottom-left) and 50% (bottom-right)) (Node 1) 83

4.16 ABOVENET: 95% Confidence Intervals for Mean Revenue for Reduced BFC plotted by the Standard Deviation of Revenue for changing demand conversion rates (CR = 50%, 40%, 30% and 20%) (Node 1) . 84
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>Architectural Transition and Integration of Cloud Services to Routers</td>
<td>87</td>
</tr>
<tr>
<td>5.2</td>
<td>CAR Architectural View</td>
<td>88</td>
</tr>
<tr>
<td>5.3</td>
<td>CAR: Peering Establishment Scenario</td>
<td>94</td>
</tr>
<tr>
<td>5.4</td>
<td>CAR: Forwarding Table Lookup Scenario</td>
<td>95</td>
</tr>
</tbody>
</table>
List of Algorithms

1. Path Inquiry Local Forwarding .............................................. 33
2. PVCR path-inquiry smart forwarding mechanism .................. 35
3. PVCR path-inquiry forwarding mechanism .......................... 36
Chapter 1

Introduction

1.1 Internet Architecture and Current Issues

As the American proverb goes, if it is not broken, why so many researchers have been trying to fix the Internet? What are the factors behind numerous calls asking for rearchitecting the Internet, the digital goose that has been laying golden eggs for the last two decades? First of all, we need to draw the fine-line between continuous transformation of the Internet trends emerging from popular applications with their underlying protocols and the proposed major architectural changes that are topic of this thesis. As we have witnessed for the last decade, the most popular applications have shifted frequently, from web surfing to peer-to-peer file sharing, and then to web-based video-sharing which is predicted to keep its leadership for a while. As for the underlying protocols which achieve connectivity among hosts wired to the Internet, early protocols developed for humble DARPA network matured into today’s modern protocol stack which is also under ongoing patches, updates and renovations to keep up with the demands and the challenges of the now highly commercialized Internet
market. However, the constant renovations on neither applications nor protocols do not stand for the major architectural changes that have been called for.

Early designers of the Internet, a.k.a. fathers and mothers of the Internet, proposed “End-to-end Argument” \[83\] which summarizes the guidelines and design principles behind the current Internet architecture. According to these principles, 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 innovation, exciting new application and protocols by ensuring low entry barrier for them. \[25,58\]. Essentially, what these principles target is to keep the Internet core transparent and simple as if its only task would be letting packets in and out \[25,58\]. This basic design choice has led the formation of the current Internet architecture which is simplistic in its core and comparatively much more complex at edge entities and functions. As envisioned by its designers, the simplistic Internet core has taken a role as a mere integrating medium which is free of constraints and requirements obligated by specific applications whereas edge entities have taken the burden of executing complex task and functions required by applications. When applications running on the edge entities are in need of advanced core functionalities, alternative solutions have been implemented at upper layers, e.g., congestion control at the network layer, overlay routing at the application layer.

The current Internet architecture can be claimed rightfully as one of the factors behind the huge success of the Internet so far. However, the very same design choice has proven to lead various issues on economics, security and routing that are painstakingly hard to solve while staying within the same design principles without making major architectural changes \[25,82\]. From economics perspective, the current
Internet architecture has led the formation of a revenue distribution structure which rewards edge entities, where most of the complex tasks are carried out and most of the innovation happens, significantly more in compared to core entities. Lacking innovation in its simplistic function set, revenues related to core services have been dropping constantly and they have been considered as commodity. Moreover, the trend of declining revenues currently coincides with the urgent need of renovation in Internet backbone as a result of annually doubling Internet traffic volume and constantly growing expectancies on reliability and overall performance of connectivity. As a result of declining revenues along with constant need for renovation of infrastructure due to growing demand, providing transit Internet connectivity at the core level is in the way of becoming an unprofitable business. Transit Internet Service Providers (ISPs) whose businesses are at the core of the Internet have been struggling to create revenue generating alternatives, economic models and innovative services to survive. ISPs have to collaborate with other ISPs in order to create such revenue opportunities since traditionally the Internet connectivity has been perceived (e.g., advertised, delivered and priced) as a point-to-anywhere (p2a) service where a customer expects its service provider to connect her to the rest of the Internet. Although the current Internet architecture has provided ISPs with exchange mechanisms to negotiate routes according to their business policies and technical priorities, these exchange mechanisms are simplistic. These existing mechanisms do not allow ISPs to control paths connecting them to Internet destinations in a flexible way as demanded by end users (e.g., enterprises, its customer ISPs).

In this work, we propose “Contract Switching Architecture” where ISPs gain the ability of expressing their demand on custom service quality and perception of expected utility (i.e., perceived value) out of demanded service for routes to particu-
lar Internet destinations. By exchanging their demand on custom services and their willingness to pay more for these services, ISPs will gain multi-hop, multi-metric negotiation abilities among each other in comparison to single metric, single hop capabilities with immediate neighbors in current Internet architecture. Such an overhaul requires to break point-to-anywhere service definitions where quality and other service parameters (e.g., price) are defined similarly for all Internet destinations into point-to-point counterparts where service parameters may be customized depending on particular Internet destinations.

1.2 Organization of the Dissertation

In Chapter 2, first, we briefly give information on the Internet architecture and economics. Then, we summarize the ongoing research on developing clean-slate architectures. Major challenge before our proposed architecture is to develop a new set of protocols which can achieve such flexibility and diversity while they ensure meeting requirements set by the Internet community for future Internet clean-slate proposals after long discussions on re-architecting the Internet [27] as transparency, diversity, security and reliability being a few. More importantly, protocols within Contract Switching Architecture should operate in a scalable manner considering the scale and complexity of the Internet. Although the point-to-point approach in service definition limits service delivery capabilities, it constitutes a simple but powerful aggregation mechanism which ensures scalable operation of routing protocols. Loosening point-to-point service definition requires us to develop innovative solutions to achieve scalability as we proposed several mechanisms such as epidemic path exploration techniques, hybrid link-state and path-vector protocols.
Then, in Chapter 4, we present our work that analyzes how much benefit can be gained both economically and technically via such decomposition and overhaul by introducing multi-hop, multi-metric negotiation mechanisms in a point-to-point service delivery definition setting. In Chapter 4, we set up a framework to measure increasing ISP negotiation capabilities for discovering end-to-end quality paths for value-added services by adopting multi-hop, multi-metric negotiation mechanisms. In Chapter 4, we also focus on risk management and economic benefit aspects of Contract Switching Architecture since we recognize them as being of the two major challenges before implementation and adoption of all clean-slate architecture proposals. Risk management and economic considerations become more important with the introduction of value-added source routing schemes, especially because these value-added services usually require ISPs to make additional investments in terms of upgrading their infrastructure, exposing their network to risks related to flexible configuration of services, reserving additional resources for service orchestration and meeting more stringent service quality requirements. Aside from these technical challenges, ISPs also take economic risks related to opportunity costs in managing their resources and allocating their existing network capacity to these services.

Delivering customized routing services and end-to-end service composition increases complexity in routing and route computation. Research efforts in developing a routing algebra and semantic mechanisms to automate configuration tasks have been targeting to mediate such complexity [105]. In addition to these efforts, integration of cloud computing paradigm into networking has given promising results for further solving complexity issues by delegating such complex routing tasks into resourceful cloud mechanisms [105]. In Chapter 5, we propose innovative solutions on such delegation capabilities as we name them “Cloud-assisted Routing”. Finally, in Chapter
6, we summarize our major findings and describe future improvements on our work.
Chapter 2

Related Work

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 Improvement Proposals

Mahajan et al. [71] 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 not compatible, both entities are better off if they negotiate, e.g. ISP A minimizes de-
lay whereas ISP B escapes overload. Negotiation through preference exchange allows pairs to find better outcomes which are not explorable using mechanisms of Border Gateway Protocol (BGP) which is the de-facto inter-domain routing protocol today.

The most exciting result of this work is that bilateral negotiation offers most benefits of global optimal routing without requiring ISPs to expose their confidential network topologies. Interesting enough, global optimal routing would end up in cases where one side of the 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. [85], introduced the 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. 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 summary, path trading and negotiation proposals show us the hidden cost of shortest-path routing and the value of negotiation.

Another backward compatible proposal with current Internet Architecture is Multi-path Inter-domain ROuting Protocol (MIRO) [109]. As default, providers learn inter-domain routes provided by BGP. To improve bandwidth or latency of default routes, or avoiding an intermediate ISP, source ISP 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 as can be seen in Figure 2.2. Bilateral path queries are similar to those in above mentioned path trading proposals. Moreover, MIRO extends bilateral negotiations by enabling negotiations with non-neighboring ISPs along the path. Alternative paths could be learned through pull based queries. Also, downstream ISPs could advertise alternative routes to upstream ISPs in a push based manner for redirecting traffic along
alternative paths. Once alternative routes are learned, 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.

Another branch of improvement proposals can be classified under the umbrella
of “Service Overlay Networks”. The basic idea behind these proposals is the separa-
tion of forwarding and routing mechanisms. Cabernet [114], Routing as a Service [62]
and Routing Control Platform [30] are some of the outstanding proposals aligned with
this approach. Even though there are major differences between these approaches,
the 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 also
follows very similar approach to define these virtual links and end-to-end contract
path composition. 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.
2.2 Clean Slate Approaches

![Diagram of NIRA's Addressing]

Figure 2.3: Taken paths are embedded in hierarchical addressing by NIRA [110]

Yang et al. [110] proposed 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 the 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 learns its provider and providers’ provider and all transit paths 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 in the routing processes in the Internet Core. For downhill segments that connects the Internet Core to destination AS, NIRA employs a DNS like ser-
vice named name-to-route lookup service (NRLS). NIRA employs provider rooted hierarchical addressing. A node address consists of two parts: i) A prefix that is a non overlapping subdivision of the provider address space, ii) Provider independent intra-domain address part which uniquely identifies the node within intra-domain network (see Figure 2.3). The first part not only addresses 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 the prefix part of the address represents alternative uphill paths to this node and the remaining part uniquely identifies the node within a specified domain (e.g. 1:1:1::1000 1:2:1:1000). A user that wants to establish an end-to-end path first chooses an uphill path using information provided by TIPP, then chooses a downhill path by NRLS queries. Once such a path established, the 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) [93] employs cone-like segmentation of end-to-end path establishment similar to NIRA. In this two-tiered model, segments representing cones are managed by a 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 a failed path.
In that sense, HLP reduces the number of route update messages significantly when 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.

Figure 2.4: HLP [93]

Feedback Based Routing [113] 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 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 a no interference rule, the chance of the concurrent failure of both transmission paths is minimized.
Chapter 3

Contract Routing Architecture

We propose an Internet architecture that allows flexible, finer grained, dynamic contracting over multiple providers \(^1\). 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 the 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 Agreements (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

\(^1\)Results presented in this chapter have been published previously at conference venues [54–57].
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 data-grams. Contract-switching introduces even more tussle points at the edge/peering points of domain boundaries by overlay contracts as depicted in Figure 3.1.

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 Quality of Service (QoS) 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 dynami-
cally 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 [112], 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 [79, 81]. 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.”

3.1 Contract Definition

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 3.2). 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 [62].

![Figure 3.2: Contract Link Abstraction](image)

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

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 3.3).

![Figure 3.3: Contract Link Abstraction](image)

**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 derivate 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

There are several types of contract links used in the definition of end-to-end paths within Contract Routing Architecture as listed below:

**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 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 3.4, following above principles, we define contract routing architecture in four modules. 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 [53] 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 3.5. This scheme more fits in large scale providers who have large number of contract links with diverse spatial characteristics.
3.2.4 Proposed Design

Exchange Module

According to measurement analysis on popular prefix destinations, these routes are mostly stable for weeks [81]. 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.
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 3.3. So, once a contract link is sold, promised tunnel should be established between ingress router and egress router.
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 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 [23], our “contract links” are between peering points within an ISP, and not between ISPs (see Figure 3.6).
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 3.6 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. \((10\text{Mb/s} \times 2\text{hrs} + 100\text{Mb/s} \times 3\text{hrs} + 60\text{Mb/s} \times 24)/($10 + $110 + $250) = 27.2\text{Mb/s/hr/$})\), 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.

### 3.4 Path Vector Contract Routing

Simply demonstrated by Figure 3.7, PVCR path exploration process starts with a path inquiry initiated by a source ISP. Path inquiry packet describes the properties (e.g., QoS, price and duration) of an end-to-end (e2e) path whose composition is desired between a source and a destination ISP. Through forwarding path inquiries,
ISPs along the way either choose to participate in composing such an end-to-end path or returning the offer by dropping (filter) the inquiry packet. The problem for intermediate ISPs is to decide whether it is technically and economically feasible to involve in such a path composition. BGP uses this type of path-vector composition process to compute end-to-end paths. But, it only considers the number of hops (i.e., ASes) a path traverses and computes the shortest-path based on a single parameter. Whereas in our case, as demonstrated in Figure 3.7, path query packet describes end-to-end path in various dimensions.

In Figure 3.7, User X (which may be a stub ISP or a company), seeks a quality path to destination ISP 5, supporting at least a guaranteed bandwidth level of 10 Mbps for 15 minutes within the total budget of $10. We consider a more general path-vector composition process where multiple metrics are used instead of just the “minimum number of hops”. Once ISP X initiates this path inquiry process, the problem for intermediate ISPs is to decide whether it is technically and economically feasible to involve in such a path composition. If they find it feasible, then they will decrement their share of budget and maybe confine the sought quality boundaries of the path query according to their infrastructure capacity and economic risks. Finally,
they have to decide which neighbors should be targeted as the next hop(s) of this path inquiry process.

Each participant ISP adds itself into the path-vector list in a received inquiry packet and forwards it to its selected neighbors. Path-vector lists kept in these path inquiry packets provide loop prevention mechanism for PVCR similarly to BGP. If path inquiry packet ever reaches destination ISP, source ISP will be informed about the e2e path with an acknowledgment packet. Finally, after reservation process, such e2e paths will be setup by stitching these forwarding tunnels (edge-to-edge links) governed by individual ISPs along the explored path.

In summary, considering i) the scale of the Internet, ii) rich forwarding options, (e.g. path explosion), and iii) dynamism of path-quality conditions, we have many non-trivial challenges for such a process namely: i) Composing an end-to-end path without requiring global coordination or knowledge of the full topology, ii) Dynamic route conditions, iii) Controlling load of inquiry packet messages, and iv) Economic risks and perception which make path composition task even harder.

In our PVCR design, each ISP only has to keep connectivity information for a bounded region. This connectivity information solely consists of peering relationships of ISPs within this region and does not include any path quality information (e.g., QoS). Furthermore, this connectivity information does not necessarily have to be up-to-date except for first hop peerings of this particular ISP. The boundaries of concerned region can be fine-tuned by individual ISPs, too. PVCR utilizes locality-awareness to reduce non-determinism in making next hop selections in path inquiry packet forwarding. If an ISP cannot locate destination ISP in his locality, then path inquiry packet forwarding will be handled by smartly-randomized mechanisms which will be described in following section.
3.4.1 Path Explosion in PVCR

As explained in Section 3.4, upon receiving a path inquiry message, recipient ISP has to decide whether it will participate in requested path composition. If an ISP does participate in this end-to-end path composition, then it has to forward this path inquiry message to a subset of its neighbors furthering the path exploration. However, this forwarding mechanism can easily lead to a flooding of path inquiries which may result in "path explosion" if not controlled. Path explosion also has been observed in other distributed path computation protocols [52]. In our early study [57], we have observed routing performance improvements achieved by the exploitation of locality information in a rather graph theoretical approach. In this work, we investigate application of ISP policies for achieving performance achievements and mitigating path explosion problem. Also, in this work, we make additional routing performance analysis of our PVCR prototype in a packet-level simulation setting.

Path Inquiry Forwarding

To control flooding of path inquiry packets, we introduce breadth (BTTL) and depth (DTTL) time-to-live parameters. BTTL parameter determines how many copies of a path inquiry message can be made by participating ISPs while DTTL parameter sets the maximum number of hops that a path inquiry can be passed forward from its source towards its destination. When BTTL value of an inquiry packet hits the minimum value of 1, this packet cannot be further forwarded to multiple neighbors, but just forwarded to a single next hop (as long as its DTTL value does not hit minimum value of 0).

To further control forwarding of path inquiry packets, we also limit the maximum number of next hops (e.g., neighbors) that an ISP can forward copies of a path
inquiry by maximum forward (MAXFWD) parameter (See Algorithm 2). As depicted in Figure 3.8, path inquiry packets are only forwarded to 3 randomly selected neighbors of total 7 neighbors due to MAXFWD limit of 3.

Although these parameters successfully prevent flooding of path inquiry packets, reachability cannot attain desired levels (i.e., over 90%) while incurring still significant cost of routing control traffic [57]. So, we introduce further improvement mechanisms on top of this randomized forwarding method.

**Locality Information**

We use local topology information to guide randomized forwarding of path inquiries, especially if they are in the vicinity of destination ISP. Also, local topology information increases scalability of PVCR by eliminating redundant copies of path inquiries in significant numbers. Local topology information here consists of the first-hop and second-hop neighbors’ ASN numbers of an ISP which can be kept by a Bloom Filter [21] in a memory efficient manner. If destination ISP is in local neighborhood, then path inquiry packets can be simply forwarded towards destination ISP without activating randomized forwarding (See Algorithm 1). In our recent work [57], we also explained how routing subproblems in local neighborhood can be solved efficiently by
a combination of local forwarding mechanisms and use of multiple layer bloom filters.

Algorithm 1: Path Inquiry Local Forwarding

\begin{verbatim}
void CheckValid(packet)
1: if packet.dTTL == 0 or ¬LoopFree(packet.Path) then
2:   Drop(packet)
3:   return
4: else if ¬LocalForward(packet) then
5:   SmartRandomizeForward(packet)
6: end if
\end{verbatim}

Policy Improvements on Forwarding

Previously, we have shown that smartly-randomized forwarding mechanisms guided by local topology information can achieve high end-to-end path exploration performance (i.e., over 95%) even at the existence of large number of non-cooperating or risk-averse ISPs. In this work, we further investigate the effects of ISP policies on PVCR routing performance.

As depicted in Figure 3.8, BTTL budget of original path inquiry is equally shared among copies of the original packet (See Algorithm 3). However, this distribution can be made more intelligently so as to favor neighbors which are closer to Internet core or which have more inter-domain connections (given these information are available). In our analysis, we employ a very simplistic policy which depends on the centrality of neighboring ISPs to distribute BTTL budget. We use betweenness centrality as a measure to determine ISP policy on forwarding path inquiry packets. Betweenness centrality is a measure which indicates how many shortest paths in a graph come to pass a particular node. An ISP can calculate an approximate value
of neighbor’s centrality by counting the number of preferred inter-domain paths, i.e., BGP AS-Path, which a particular neighboring ISP appears. Instead of using betweenness centrality, an ISP may deploy an alternative policy based on the volume of inbound traffic forwarded by its neighbors. Here, we only aim to investigate improvements introduced by even simple ISP policies on path exploration instead of specifying a particular policy.

**Experimental Setup**

For our analysis, we implement PVCR forwarding mechanism guided by both local topology information and ISP forwarding policy. Then, we investigate routing performance of PVCR on recent AS-level topology provided by CAIDA as of January 2010 [8]. We randomly choose 10,000 ISP source-destination pairs out of 33,508 ISPs, execute PVCR with these selected pairs and repeat the experiment with 31 different seeds. To measure effectiveness of local information and forwarding policy improvements, we define 6 different scenarios: i) *No Locality (N)* scenario: no topology information is provided to PVCR, ii) *Locality (L)* scenario: PVCR is guided by local topology information, and iii) *Locality with Filtering Policy (LP)* scenarios, path inquiry packets are forwarded in guidance of local topology information, however they are filtered (or dropped) by intermediate ISPs at different rates according to ISP filtering policy. For example, in LP50 case, path inquiry packets are dropped by an intermediate ISP with 50% probability. By introducing these filtering probabilities, we want to emulate ISP policies which reflect risk averse or non-cooperating ISP behavior. Also, intermediate ISPs may drop path inquiry packets due to unavailability of edge-to-edge resources (e.g., bandwidth), or infeasibility of an offer (e.g., low price), or simply rate limiting policies imposed on path inquiry packets.
Algorithm 2 PVCR path-inquiry smart forwarding mechanism

MaxFWD - parameter limiting max. # of packet copies

Policy - preferences of an ISP while selecting next-hop for an inquiry packet, i.e., neighbor degree of connectivity, tier, business relationship.

GetNeighbors(N,M,Policy) - returns most preferred N neighbors skipping first M entries following a policy ordering, returns empty list if there is no neighbor left not listed before

Feasible(Packet,Neighbor,Policy) - returns a list of terms related to path establishment, e.g., price requested, QoS parameters, duration etc., returns empty if providing this service is not feasible

void SmartRandomizeForward(packet)

1: numofcopies ← 0
2: allowance ← min(packet.bTTL, MaxFWD)
3: selectedneighbors ← ∅
4: S ← ∅
5: while numofcopies < allowance do
6:     S ← GetNeighbors(allowance, allowance − numofcopies, Policy)
7:     if S == ∅ then
8:         return
9:     end if
10:    for all neighbor : neighbor ∈ S do
11:        if Feasible(packet, neighbor, Policy) == ∅ then
12:            return
13:        else if LoopFree(packet.path ∪ neighbor) then
14:            selectedneighbors ← selectedneighbors ∪ neighbor
15:            numofcopies = numofcopies + 1
16:        end if
17:    end for
18: end while
19: if numofcopies > 0 then
20:     ForwardPackets(packet, selectedneighbors, numofcopies)
21: end if
Algorithm 3 PVCR path-inquiry forwarding mechanism

```c
void ForwardPackets(packet, selectedneighbors, numofcopies)

1: packet.bTTL = packet.bTTL/numofcopies
2: packet.dTTL = packet.dTTL − 1
3: packet.Path ← origPacket.Path ∪ self
4: for all neighbor: neighbor ∈ selectedneighbors do
5:   packet.Terms ← Feasible(origPacket, neighbor, Policy)
6:   send(neighbor,packet)
7: end for
```

3.4.2 Routing Performance and Scalability Analysis

Path Exploration Success

As shown in Figure 3.9, PVCR achieves nearly 100% path exploration success for all L scenarios with various BTTL and MAXFWD values. For N and LP scenarios, path exploration success significantly improves with increasing BTTL and MAXFWD values. Moreover, even for 50% inquiry packet filtering, i.e., LP50 at BTTL=4096 and MAXFWD=16, PVCR achieves 90.24% path exploration success rate. These results indicate significant improvements over our previous work which lacks ISP forwarding policies, i.e., BTTL distribution according to neighbor centrality. In fact, PVCR is able to perform 62% path exploration success for LP50 scenario given same BTTL and MAXFWD parameter values in our previous study [57].

Path Diversity and Path Stretch

We also investigate average number of explored paths for each path inquiry process so as to measure path diversity captured by PVCR. As shown in Figure 3.10, PVCR is able to provide end-to-end paths in great numbers. In that sense, we can argue that PVCR is capable of capturing rich path diversity offered by Internet as discovered by previous research [94]. Similar to path exploration success, path diversity improves
with the increase in BTTL and MAXFWD values. However, in compared to our previous analysis, inception of centrality measure in ISP forwarding policy reduces the number of explored paths. This decrease can be attributed to bias that favors more central ISPs over low tier ISPs. Consequently, path exploration is constrained into a tier-based, relatively structured fashion which does limit exploration of peering between ISPs of same tier in compared to unbiased case.

Another indicator of routing performance is path stretch which measures the
ratio of length of discovered paths to the length of shortest path. With the increase in BTTL and MAXFWD parameter values, PVCR becomes more capable of discovering shorter paths due to increase in path inquiry packet limits as can be seen in Figure 3.11. Although application of ISP forwarding policies may result in slight decrease of the number of explored paths, there is no performance degradation in terms of path stretch in compared to our previous study [57]. Path stretch values between 1.5 and 2 indicate that PVCR is not only able to find great number of paths but also paths with high quality. Considering the path inflation due to BGP policy-based routing, PVCR path stretch performance can be considered as comparable, if not superior to BGP path stretch performance.

**Control Traffic**

In Figure 3.12, we plot how many copies of original path inquiry packet is made on average. In Figure 3.13, we plot how many times path inquiry packets are forwarded on average. These two figures together reflect the performance of PVCR path explosion prevention mechanisms. By comparing N and L scenarios, we see that locality information significantly decrease routing control traffic (nearly by ten-fold...
for BTTL=4096 and MAXFWD=16). As expected, routing control traffic increases with the increasing BTTL and MAXFWD parameter values. However, even at the worst case of BTTL=4096 and MAXFWD=16, for L scenario, PVCR generates 600 bps of non-priority control traffic (assuming 200 bytes of median discovery packet size [3] during 5 minutes of path exploration expiry period). We believe that 600 bps of control traffic per flow is reasonable considering the target of elephant end-to-end QoS flows within enhanced end-to-end service delivery context.

3.4.3 Traffic Engineering with PVCR

Traffic Engineering (TE) has still been considered as a challenging research area by many, however well studied [44,88]. In part, this challenging characteristic can be attributed to multi-dimensional (e.g., intra- and inter-domain), multi-party (e.g., competing and cooperating ISPs) aspects of traffic engineering [14,43,71,77,99]. Adding further to its complexity, traffic engineering can be also defined as a multi-metric optimization problem (e.g., congestion avoidance and cost minimization) [14,37,44,96]. Due to this complex nature, researchers have developed different perspectives for
approaching this large set of problem space.

From a single ISP perspective which takes inter-domain routing priorities first, traffic engineering problem can be defined as improving network performance metrics, such as congestion on inter-domain links, or inter-connectivity costs while minimizing the global effects of TE policies on the Internet (e.g., minimizing BGP updates or route flapping) [77, 96]. From a more inclusive perspective which combines intra- and inter-domain views of an ISP, traffic engineering translates into configuring network so as to minimize policy fluctuations and counterproductive interaction between intra- and inter-domain routing [14, 44]. From a game theoretical approach, traffic engineering is an ongoing negotiation on route selection between neighboring ISPs, customers and content providers [51, 71, 88]. Finally, from a protocol-wise approach, traffic engineering is simply computing inter-domain paths with mutual agreement in a distributed manner [52, 109, 110].

Path-Vector Contract Routing (PVCR) introduces limited control traffic costs and new flexibilities, especially in ISP representation, routing metrics definitions and route computation so as to overcome limitations of previous methodologies while
preserving their merits. First, “set of links” representation of ISPs results in finer-granularity of path negotiation and increased path diversity as also targeted by previous proposals [35, 109]. Announcement of path-vector contracts for advertising edge-to-edge QoS enabled services on these links extend the bilateral path trading [85] and path negotiations [71] into a multi-hop, multi-metric path form. Path-vector style path computation mitigates scalability issues in extending these bilateral path trading approaches into multi-hop path negotiations due to explosion of alternative paths among multiple ISPs. At the same time smart-randomized forwarding of path inquiries and ad-hoc path computation mechanism alleviate the scalability and AS-path exploration issues in distributed path computations methods introduced before [52].

Although PVCR introduces extra path-inquiry traffic, this routing control traffic can be managed by introducing limits as shown in Section 3.4. Current best practices of inter-domain traffic engineering also cause extra control traffic (e.g., BGP updates, route flapping, OSPF weight adjustments, and even route instabilities). Due to explicit multi-metric, multi-hop negotiation of end-to-end paths, PVCR eliminates route instabilities.

From routing economics perspective, PVCR allows an ISP to advertise its edge-to-edge services in different prices independent of each other by means of “set of links” abstraction of ISP. In current point-to-anywhere SLA settings, this flexibility either does not exist or is implied in a limited fashion. Moreover, since the price is considered as an explicit routing metric, price information itself becomes an inherent part of traffic engineering as targeted by many previous studies [37]. To observe PVCR traffic engineering capabilities, we have developed a prototype of PVCR as an overlay to BGP protocol on SSFNet framework [90] which provides BFC compliant implementation for BGP-v4 [80] as a packet level simulator.
Prototype Implementation

Our prototype implementation of PVCR protocol consists of following modules:

**Resource Manager** is responsible for reserving and releasing edge-to-edge capacity during end-to-end contracted path establishment operations. Given an intra-domain topology, resource manager allocates min-max fair share bandwidth capacity for each edge-to-edge link if these virtual links have common physical links. Resource Manager also coordinates resource allocation operations involving multiple routers. Resource Manager regularly tracks utilization values of edge-to-edge links for the use of congestion-aware pricing.

**Traffic Manager** is responsible for holding simplified traffic matrices that we use for our simulations. In regular intervals, Traffic Manager keeps track of traffic demand and generates path inquiry packets to explore end-to-end paths which can meet this demand. In our simulations, we have used fixed synthetic traffic volumes. Although we have also used traffic values calculated according to Gravity Model [74] considering populations of cities which routers in Rocketfuel topologies reside, we only share reports on synthetic traffic scenario for the purposes of demonstration.

**Session Manager** is responsible for forwarding path inquiry messages in accordance to ISP policies. Session Manager also applies ISP strategy (policy) by accepting (or rejecting) to participate in path establishment offered by these inquiry messages. Session Manager considers resource availability, requested QoS parameters and price offered by these path inquiries for making these decisions.

Experimental Setup

In our first set of experiments, we have used a simplistic 10 by 10 grid topology at inter-domain level. There are 100 ISPs and 370 inter-domain links in our grid
topology. Totally, there are 4248 edge-to-edge links. For our grid topology, all inter-domain links have 100 Gbps bandwidth capacity. Also, all topologies have 100 Gbps edge-to-edge capacity between border routers. We have a traffic flow between every ISP source and destination pair in our topology which is total of 9900 traffic flows. We set MAXFWD parameter as 2 and BTTL parameter as 1024. Our contract routers operate at 30 minutes intervals to update their traffic demand matrices and initiate path query messages. Border routers try to establish end-to-end paths within 30 minutes frame in according to these traffic matrices. Also, contract term, which determines the lifetime of a constructed end-to-end path, is set as 30 minutes. Each border router waits for 10 minutes before initiating another path exploration process (up to 3 trials) if it does not receive any response to previous path inquiries.

To create a set of benchmark scenarios, we measure required traffic flow volumes so as to achieve particular link utilization levels at the mostly loaded inter-domain link of the topology. For these benchmark scenarios, traffic flows take the end-to-end paths calculated by BGP protocol. In an example, for the case of BGP50, volumes of traffic flows are fixed at 115 Mbps so that the mostly loaded inter-domain link of the topology would operate at 50 percent link utilization. Similarly, for the
case of BGP100, size of traffic flows are fixed at 231 Mbps so that the mostly loaded inter-domain link of the topology would operate at 100 percent link utilization. Then, with traffic flow size determined for each scenario, we run PVCR prototype for 90 hours. We have used congestion-aware price function depicted in Figure 3.14.

**Evaluation**

We group unidirectional edge-to-edge links according to their egress inter-domain link and take the average service price. We plot these average service price of individual
Figure 3.16: Network Price Map on a 10x10 Grid Topology
ISPs on surface plot by again taking average of the prices of all inter-domain links belonging to each ISP. We also plot service price maps for paths calculated by BGP for comparison. For example, in Figure 3.15(a) and 3.15(c), size of all traffic flows are set at 155 Mbps. Traffic flows follow paths calculated by BGP in Figure 3.15(a) and follow paths calculated by PVCR in Figure 3.15(c).

In Figure 3.15, we plot price maps for the scenarios where maximum loaded inter-domain link of the topology operates at 50, 70, 90 percent of utilization. As utilization level increases, hot spots appear to arise for topologies managed by BGP. Meanwhile, PVCR achieves price stability and greater egress traffic load balancing as depicted by equally elevated flat surface of price maps for all utilization levels. For topologies managed by BGP, service prices peak at 28 - 30 level for central ISPs which constitute the hot spots, while periphery ISPs operate at relatively lower price range of 20 to 24. However, for topologies managed by PVCR, prices are homogeneous within the range of 20 to 21.

Another interesting observation is that even periphery ISPs of BGP managed topologies operate at relatively higher utilization levels in compared to their counterparts in PVCR managed topologies. As a result of its smartly-randomized forwarding of path inquiries and multi-metric negotiation mechanisms, PVCR achieves highly successful egress load balancing performance. Similarly, by charging higher service prices or filtering path inquiry packets for its highly utilized edge-to-edge links, an ISP can explicitly manage ingress load balancing.

In Figure 3.17, we plot price maps for scenarios where mostly loaded inter-domain link of the topology, which are managed by BGP, cannot deliver all the traffic packets to next hop due to congestion. For these scenarios, PVCR continues to deliver all packets successfully. However, as the utilization levels increase, flat surface
observed at lower utilization levels appear to become slightly disturbed at the edge of the topology. This disturbance can be attributed to less number of inter-domain connections of periphery ISPs which limits the load balancing capabilities of PVCR at the edge of the grid topology. Yet, for the topologies managed by PVCR, service price differences never exceeds 0.2% between periphery and central ISPs.

We depend on default shortest-path calculation and tie-breaking mechanisms of BGP protocol. This simplistic approach does not include any traffic engineering policies which have been adopted widely by today’s ISPs. However, regardless of the
effectiveness of these TE practices, BGP is prone to create hot spots [40] due to lack of coordinated TE mechanisms with rich expressive capabilities. Although an ISP may selfishly improve routing performance for a while as a result of its TE efforts, it cannot coordinate ISPs beyond immediate neighbors to sustain attained network performance due to lack of multi-hop negotiation capabilities of current Internet architecture. Aggressive traffic engineering policies which do not consider counter-effects of route flapping and frequent switching between egress-paths have been shown to lead routing instabilities. In that regard, our BGP scenarios are instrumental to reflect shortcomings of current traffic engineering policies in their simplistic form.

3.4.4 PVCR Performance on PoP-level Topology

Selection of topology can significantly affect the outcome of performance evaluation for a routing protocol [45]. Concerns for possible bias related to topology are especially valid for inter-domain routing protocols since concerned topologies face the challenge of modeling the Internet.

ISPs locate their routers at major cities where large volume of Internet traffic originates from metropolitan areas. These locations are commonly called as points-of-presence (PoP) for a service provider. A PoP-level topology represents an ISP\textsuperscript{2} as a set of routers in geographically separated PoPs as opposed to AS-level topology where an ISP is modeled as a single router. Hence, two neighboring ISPs connected at several PoP locations via multiple inter-domain links, are only represented as two routers connecting via single link in AS-level granularity. To address possible shortcomings of AS-level topology, it is necessary to verify our previous simulation results also in PoP-level granularity.

\textsuperscript{2}By "ISP", we implicitly refer to a single ISP domain with the convention that an ISP may have multiple such domains.
Topology Generation

To generate PoP-level topology, we use iPlane dataset [70] retrieved on June 30, 2012. iPlane dataset provides clusters of IP addresses identified by grouping algorithms and techniques based on packet traces as described in [70]. iPlane dataset also provides latency and packet loss rate measurements on links connecting these clusters. Using CAIDA AS-ranking dataset provided in [12], we selected top 50 ISPs. From iPlane dataset, we only extract clusters and links belonging to these ISPs. Then, we removed the islands which are the disconnected components of the resulted graph.

To extract a useful topology, we need to aggregate clusters into PoPs as these clusters may belong to same PoP locations. We followed aggregation algorithms given in [31,86]. According to the technique described in [31,86], we first aggregate clusters if the measured latency between two clusters is less than 5 ms and are not separated by a distance more than 20 miles as resolved by MaxMind geolocation database [49]. If the locations of more than half of the IP addresses belong to two cluster sets agrees, then two clusters are aggregated as a single PoP. Aggregation process resulted in a topology consisting of 1,238 PoPs, 4,130 inter-PoP links. Of these 1,238 PoPs, 696 PoP locations are neighboring with more than one ISPs and there are 702 inter-domain links connecting these border PoPs.

Experimental Setup

In our experimental setup, we have 10,000 traffic flows between selected PoP destination-source pairs. To calculate volume of traffic belong to source-destination PoP pair, we have used the Gravity Method [65]. The Gravity Method considers populations of cities where these PoPs are located and generates traffic proportional to the magnitude of these populations multiplied with each other similar to how magnitude of
gravity force between heavenly objects calculated with respect to their body masses.

To resolve populations of PoP locations, we have used MaxMind GeoIP and MaxMind WorldCities databases [49,50]. First, we have picked 3 IP addresses belonging to a PoP and then verified the resolved city location of these 3 IP addresses. If they agree on the same city location, then we mark the PoP as usable during traffic flow generation. Otherwise, we do not use this PoP. Out of 1,238 PoPs, we were able to resolve population of 681 PoPs (55%) using MaxMind WorldCities databases [49,50].

To calculate required traffic loads to achieve certain traffic utilization as described in Section 3.4.3, we run a preliminary simulation where each border PoP location is represented by an OSPF-BGP running router. In this preliminary topology, all intra-PoPs are eliminated. However, to preserve intra-domain topology for hot potato routing decisions taken by BGP, we set the interface cost of OSPF interface connecting two border PoPs as the number of intra-PoPs connecting them in original topology. Also, to preserve the total latency between routers, we modify the latencies of inter-PoP links accordingly. Then, we calculate normalization factors to be used in our simulation for achieving traffic utilization varying between 50% to 120%.

Evaluation

According to various traffic scenarios where the most loaded inter-domain link of the topology is running at 50% to 120% traffic utilization, we calculated the average price of bandwidth advertised by border routers. As described in previous section, we group unidirectional edge-to-edge links according to their egress inter-domain link and take the average service price. We plot these average service price of individual ISPs on surface plot by again taking average of the prices of all inter-domain links.
belonging to each ISP. We also plot service price maps for paths calculated by BGP for comparison.

As nearly all ISPs are connected to each other at several PoP locations, it is challenging to represent this topology on a planar surface. To decide how to place ISPs on a planar surface and their relative locations with each other, we have placed the ISP with the highest average price on the center of price map. Then, we have placed other ISPs in clockwise order with respect to their discovery order by depth-first search algorithm originating from this center.

As seen in Figure 3.18, hot spots appear on BGP managed topologies whereas CR managed topologies are evenly elevated. Also it is worth noting that overall average prices are significantly lower in CR managed topologies thanks to superior traffic engineering capabilities of PVCR.

In Figure 3.19, we reported the average bandwidth price for higher utilization levels. Similar to PVCR performance at AS-level topology, PVCR is able to deliver more traffic than BGP whereas BGP managed topologies suffer traffic congestion at appearing hot-spots.

Gravity method generates significantly different distribution of traffic flow sizes in comparison to uniform traffic flow size distribution used in previous section. This power-law obeying traffic distribution is the main factor behind more disturbed level of price map surface reported in Figure 3.19 in comparison to Figure 3.17. However, achieved PVCR performance on PoP-level topology are very similar to results on AS-level topology as reported in previous section. This similarity of performance of PVCR protocol on a finer-grained PoP-level topology successfully address the possible bias concerns related to shortcomings of representation in AS-level topology.
(a) BGP with 50% highest link utilization  
(b) PVCR with 50% highest link utilization

(c) BGP with 70% highest link utilization  
(d) PVCR with 70% highest link utilization

(e) BGP with 90% highest link utilization  
(f) PVCR with: 90% highest link utilization

Figure 3.18: Network Price Map on a PoP-level iPlane topology at lower utilization levels
Figure 3.19: Network Price Map on a PoP-level iPlane topology at higher utilization levels
Chapter 4

Benefit Analysis

4.1 Technical Benefit Analysis from Network Perspective

Today, delivering value-added services, which involve interactive technologies, infrastructure leasing and rich media content, has become more central to create revenue sources for Internet Service Provider (ISP) companies [46]. These opportunities come with their own challenges of demanding service performance levels and effective management of ISPs’ networks. In the face of these challenges, ISPs exercise more aggressive traffic engineering, multi-homing and peering strategies while upgrading their own networks into more resourceful and intelligent domains. However, ISPs are constrained by several factors in their efforts at improving service delivery performance. First of all, an ISP has to minimize cost of service delivery by applying routing policies to avoid expensive transit links even when they are more preferable from a pure technical perspective [47]. Widely applied “Valley-free Routing” significantly limits sharing of these valuable connections among peers. Moreover, “Hot-potato Routing” or “Early Exit” policies are typically deployed to achieve relative low network utilization by forwarding transit traffic to the earliest possible exit at the cost of latency.
Fortunately, ISPs have many alternatives to mitigate such limitations. Through bilateral negotiation of paths via direct peering, an ISP can achieve more control over how its traffic is routed within the domains of its neighbors in according to “Cold-potato Routing” or “Best Exit” strategy [47, 89]. Newly emerging partial-transit, paid-peering mechanisms and n-valley exceptions to no-valley rule now allow more flexibility for utilizing transit connections to the popular destinations for partial cost-recovery of establishing these valuable peering relationships [29]. Broadening geographical coverage through establishing additional points-of-presences (PoPs) is also an option for improving performance of service delivery. Although effective locally, these practices have limited influence on routing decisions beyond border of immediate neighbors (or in some cases beyond two-hop locality). This limitation can be mainly attributed to the lack of effective signaling protocols for ISP collaboration and limited choice of expression, i.e., single-metric single-best path routing.

An increasing number of clean-slate Internet architecture proposals have advocated for the introduction of more flexible signaling mechanisms in various forms such as source-based routing schemes, path query mechanisms, or multi-path approaches [35,93,109,111,113,114]. Nevertheless, it can be argued that they all serve a common purpose of extending bilateral negotiations between neighboring ISPs into multi-hop, multi-metric form in a broader context than local improvements.

Here we analyze how effective current local improvement mechanisms are in comparison to multi-hop, multi-metric negotiation techniques introduced by these clean-slate proposals. More specifically, we pose the question: “How can service performance be improved by an ISP with its increasing reach of wide-area route control capabilities?” Our analysis on Internet traffic traces show that multi-hop,
multi-metric negotiation mechanisms can achieve significant success with 50% more performance than what is currently offered by localized single-hop mechanisms. Further, we aim to find the optimum “negotiation diameter,” which suggests the number of parties participating in achieving desired performance gains. Our initial analysis reveals that the negotiation diameter is comparable to the diameter of the Internet, i.e., 4.52 ISP as of 2008 [10]. These results indicate a significant need for development of automated SLA mechanisms so as to manage scale and complexity of extending the existing bilateral negotiations into multi-hop ones. We believe that our analysis, in simple yet realistic framework, serves as an effort to inform recent discussions on the design of next generation routing protocols.

4.1.1 Effectiveness of Current Routing Policies

Access Level Choice Expression

Today, a typical ISP forwards traffic to the closest exit point of its domain so as to minimize resource usage in its own network. Many open peering agreements confirm the significant implementation of this practice called “Hot-potato Routing” despite of its well-known handicaps in achieving overall end-to-end service quality [95]. As shown in Figure 4.1, ISP S, usually does not have a chance to influence ISP 1’s decision on choosing exit points for traffic destined to ISP D. Even though it is possible to implement “Cold-Potato Routing” or “Best Exit Strategy” practices to achieve better quality at next-hop level in paid settings, ISP S cannot express this choice beyond ISP 1 which is the immediate next-hop neighbor of ISP S.

Service Quality Level Choice Expression

Current service level agreements between ISPs promise soft-guarantees over latency and packet loss rates over a single-hop only. ISPs regularly monitor end-to-end quality
of the paths in terms of packet loss and latency so as to measure partners’ compliance with these promises. However, these promised performance levels are often coarse-grained and prone to temporal changes. More reliable service conditions can be offered within domain borders but hardly beyond these borders. In our scenario, we depict a very simplistic case where an enterprise ISP tries to achieve more reliable, better quality paths by stitching these one-hop, intra-domain guaranteed paths offered by local providers. Although very simply put, this scenario reflects the current position of an enterprise ISP, which leverages its spatially-constrained view to make local improvements so as to achieve greater service quality.

4.1.2 k-step Local Improvement Policy

An ISP may establish paid peering agreements with alternative providers to detour local performance bottlenecks. Also, an ISP can improve quality in provided services for a geographic region via broadening its points of presence (PoP) locally. We classify these localized methods as “k-step Local Improvement Policy”, where an ISP extends its reachability horizon and improves the service quality by getting better connected to the rest of the Internet through more diverse set of connections via local peering and enhanced presence. According to this model, an ISP is able to choose optimal paths within k-hop locality. Beyond k-hop neighborhood, traffic will be forwarded according to the traditional hot-potato routing. In other words, k-step local improvement policy allows an ISP to utilize k-hop source-based routing
for detouring local performance bottlenecks. Our intention is to quantify overall benefits that can be ripened by these local policies against performance of multi-hop negotiation methods proposed by clean-slate architectures which offer end-to-end quality paths comparable to global optimum [71].

To make a fair comparison of the k-step local improvement policy and clean-slate proposals, we model realistic Internet policy practices of today. Many researchers have proposed several models to capture such practices [22,33,69,70,76,78,98]. Among these models, we choose to work with (min AS Path, closest exit) model as a base case for our evaluation. Minimum AS Path model represents the selection of shortest end-to-end path in terms of intermediate ISP count along the path. Closest Exit model represents the “Hot Potato Routing” policy which was explained earlier. As reported by Madhyastha et al., these two policies together are capable of predicting 65% of the current Internet routes correctly. For our analysis, we have implemented (min AS Path, closest exit) and k-hop local improvement policies by using the JUNG library [67]. We would work with more accurate models which can predict Internet routes more accurately as they are more capable of modeling current policy practices [76,78]. However, shortest path and hot potato routing policy combination is better suited for our purpose of representing a base case. In our base case, k-hop local policy improvements are not shadowed by the structure of the Internet, which has been already deformed by traffic engineering policies.

We use the iPlane PoP-level Internet topology which provides latency values and packet loss rates measured for the links connecting ISPs’ Points of Presence (PoP). In [69], the authors have showed that these measured values have been stable after many repetitions. We construct a realistic Internet topology which is closer to the hierarchical model as described in [33]. Although flattening Internet becomes a
reality [34,39] and our model is prone to measurement bias as any topology-based study is [39], our topology model is still able to capture most of the Internet routes effectively [69].

4.1.3 Policy Performance Evaluation

For our analysis, we randomly picked 101,000 source and destination PoP pairs for our end-to-end path calculations, i.e., ISP 1-PoP1 as source and ISP 2-PoP 2 as destination. Then, we constructed the end-to-end paths according to 1) (min AS Path, closest exit) policy and 2) k-hop local improvement policy for different k values. As also explained in previous section, for the second case, we combine the best possible path segment within the k-hop locality with the path segment which is calculated according to (min AS Path, closest exit) policy combination after the k-hop border.

Overall Improvements

We quantify the improvements in latency and packet loss gains achieved in comparison to the base case (min AS Path, closest exit) policy. In Figure 4.2, we plot the improvements achieved by k-step local improvement policy in increasing k values. So, an ISP, who is able to choose best path within 2-hop locality, can improve its end-to-end connectivity 18.21% in terms of latency and 35.31% in terms of packet loss on average. For 4-hop locality, on average local improvement gains are 42.51% for latency and 63.01% for packet loss. Overall, maximum improvement, i.e., 70.16% for latency 88.18% for packet loss rate, can be achieved via 13-step locality improvements.

In Figure 4.3, we normalize the k-step improvements with global maximum improvement values in order to analyze the contribution of each step increase in value of k, i.e., each step expansion in an ISP’s horizon of negotiation. Latency improve-

\footnote{Results presented in this chapter have been published previously at conference venues [54,55].}
Figure 4.2: Average routing performance improvements of k-step local improvement routing over hot-potato routing (hops as PoPs)

Figure 4.3: Normalized average routing performance improvements of k-step local improvement routing over hot-potato routing (hops as PoPs)

...ments seem like to be spread more evenly with the increase in k, whereas packet loss improvements are greater at 2-hop locality and getting smaller with increasing range of ISPs’ horizon. Improvements at 8-hop are still significant with nearly 10% of all improvements achieved for both metrics. However, beyond 8 hops, improvements become negligible.

Overall, with 2-hop local improvements, where an ISP establishes any direct peering within 2-hop locality to greedily improve its upstream connectivity, only 25% of the maximum available improvement can be achieved for latency. This value is 40% for packet loss rate. To achieve 90% of the overall possible improvements offered by the underlying network structure, an ISP should seek improvements beyond its 6-hop...
locality. These results clearly show that 1-hop, 2-hop ISP local improvements through bilateral negotiations are not able to adequately capture performance improvements which are offered by diverse Internet paths.

**Variation with End-to-End Path Length**

To analyze the effect of varying path length on improvement performance of k-step local improvement policy, we separately observe the performance gains on end-to-end paths with varying path lengths. Such analysis reveals us if the k-hop negotiations are improving the end-to-end paths differently according to their lengths. Due to enormous number of all possible source-destination pairs among PoP points, which is nearly more than 10 billion pairs, we had to work with a sample size of 130,000 randomly selected among all possible pairs. Constructed paths between these source-destination PoP pairs are of various lengths.

In Figure 4.4, we plot improvements for latency. As the length of the end-to-end path increases, the maximum available latency improvement does increase correspondingly. This result is not particularly surprising due to physical characteristics of communication that results in greater latency as the end-to-end path becomes longer. As the length of end-to-end path increases, more alternative paths become available due to the explosion of possible intermediate nodes between source and destination. On diverse Internet market, increasing number of alternatives gives rise to greater improvements for latency.

In quite contrast with latency metric, packet loss rate improvements decline as the length of end-to-end path increase as can be seen in Figure 4.5. This result simply reflects the increasing probability of a packet being dropped as the path traveled by this packet gets longer, again not surprisingly. However, in contrast to latency, explosion of path alternatives with increasing path length does not help as much for
overcoming packet losses.

The variances of improvements achieved on different path lengths and our random path sampling method may raise a question of bias in measured improvements, such as: “Does greater representation of n-hop long paths result in capturing improvements on n-hop long paths better which may create a bias in our analysis?”.

However, as also depicted in Figure 4.4 and Figure 4.5, there are no direct relationship observed between number of paths of varying lengths and improvements achieved on these paths on average.

**Variation with Finer-Grained Path Length Classification**

In Figure 4.6, we separately plot k-step local improvement policy performance with increasing values of k for each class of end-to-end path length for the latency
Figure 4.6: Latency improvements of k-step local improvement routing over hot-potato routing for various path lengths in # of PoPs

Figure 4.7: Normalized latency improvements of k-step local improvement routing over hot-potato routing for various path lengths in # of PoPs

As a result of calculating shortest path with respect to multiple metrics, we see that 4-step local improvements can still improve latency on 2-hop long paths. For a 2-hop path, 4-step local improvement can come up with a 3-hop or 4-hop long alternative paths which are longer in hop count but “shorter” in latency. Another interesting result is that the gains in latency for 2-step and 4-step improvements are significant for paths 7-hop and shorter. After this point, contribution of 6-step and beyond 8-step local improvement policies become dominant. Figure 4.7, reflects similar results in normalized values.

In Figure 4.8 and Figure 4.7, we separately plot k-step local improvement policy performance with increasing value of k for each class of end-to-end path length for packet loss metric. Here, an interesting result to be observed is that for all 2-hop long
Figure 4.8: Loss rate improvements of k-step local improvement routing over hot-potato routing for various path lengths in # of PoPs

Figure 4.9: Normalized loss rate improvements of k-step local improvement routing over hot-potato routing for various path lengths in # of PoPs

paths, packet loss can be avoided completely by adopting 2-step local improvement policy. In spite of decrease in their contribution for longer paths, 2-step and 4-step local improvements still keep their significant share for paths 11-hop or shorter.

**Diameter of Negotiation**

In Figure 4.10, we plot the number of ISPs which are required to cooperate so as to establish end-to-end optimum paths. On average, approximately 5 ISPs need to be involved in such path negotiations to achieve the highest performance (4.998 and 4.733 ISPs for optimum packet loss and latency, respectively). Although relatively short, diameter size of 5 clearly emphasizes the necessity for the development of automated
Figure 4.10: Number of ISPs on the Global Optimum Path According to Latency and Loss Rate Improvement Policy

SLA mechanisms in achieving desired multi-metric, multi-hop negotiation. Also, in Figure 4.10, we plot the difference in path lengths between current best paths and optimum paths as error bars. Well bounded differences confirm the lack of effective signaling mechanisms which allow ISPs to construct end-to-end paths via negotiation on multiple metrics, i.e., AS-path length and latency.

**Evaluation Summary**

We can summarize our evaluation with the following key insights:

- A minimum of 6-step negotiation is necessary in order to leverage most (i.e., 90%) of the path diversity available from the Internet topology.

- 1-step and 2-step improvement policies provide good instruments to improve paths shorter than 5-hops.

- Number of ISPs need to be involved in end-to-end path negotiations is approximately 5 on average.

Today, improvement policies, which involve 1 to 4 PoPs, can be achieved through bilateral negotiations. These policies are the most adopted improvement
methods since they can be easily implemented via current BGP mechanisms, e.g., Multi-Exit Discriminator, MP-BGP and GMPLS [15]. Moreover, these improvement policies provide good instruments to improve paths shorter than 5 hops. Nevertheless, they fall short of taking full advantage of the diversity on the Internet paths, especially for the paths longer than 5 hops. For these paths of length 5-hop or longer, which are comprising more than nearly 89% of the our samples of Internet paths, 1-step to 4-step local improvement policies can deliver 50% or less of what are offered by the best alternative paths, considering both performance metrics.

4.1.4 Implications for Clean-Slate Internet Architecture Proposals

Previous analysis at intra-domain level routes has revealed that a significant majority of the intra-domain paths connecting ingress-egress edge routers have more than one disjoint path alternatives [94]. Similarly, even at the inter-domain level, 75% of the time, there can be found disjoint end-to-end paths. Furthermore, several studies have shown that large number of paths on the Internet (up to 80%) have alternative paths that offer better quality in terms of delay, packet loss or bandwidth [84, 91]. These findings all together indicate a major mismatch between available resources and offered connectivity services. In our work, we investigate how increasing wide-area control capabilities mitigate this problem so as to understand the extent of the needed ISP collaboration.

An increasing number of researchers proposed involvement of negotiation into routing [71, 85]. Mahajan et al. [71] propose Nexit (Negotiated Exit) Framework for negotiating inter-domain paths taken by traffic flows originating from neighbor ISPs. Nexit Framework is based on bilateral negotiations taking place between directly
connected ISP pairs who exchange their preferences on operational issues such as which traffic flow should traverse which inter-domain link connecting neighbors. Even for the cases where ISPs’ incentives are conflicting (e.g., ISP A targets to minimize delay whereas ISP B chooses to decrease link utilization) both entities are better off if they negotiate. The most exciting result of this work is that bilateral negotiation offers most benefits of optimum case where an imaginary authority with complete information decide on the optimum paths solely on technical terms regardless of negotiating parties’ benefits. Another similar work by Shavitt et al. [85], introduced the term of bilateral “path trading” between neighboring ISPs in a similar bargaining scheme. In our work, we consider multi-hop negotiation on explicit performance metrics shared by all negotiating parties instead of bilateral negotiations. We also investigate the diameter of negotiations, i.e., 5 ISPs, which implies the high cost of bilateral path-trading mechanisms in comparison with multi-hop path negotiations on explicit performance metrics for achieving comparable performance improvements.

Several researchers have proposed source-routing schemes to solve various issues of the current Internet routing architecture [35, 93, 109, 111, 113, 114]. Extending the idea of bilateral negotiation between neighboring ISPs, Multi-path Inter-domain Routing (MIRO) allows ISPs to request alternative routes from their upstream providers so as to increase inter-domain path diversity. So, an ISP can utilize this flexibility to avoid high-latency links along the way or to avoid a competitor ISP [109]. In a recent work of Su et al. [92], authors show that such reroutes around bottleneck links can achieve even greater performance improvements guided by mechanisms similar to Content-Distribution Network redirections. Overall improvements achieved in our work also confirm the promising aspects of increased diversity via such path-query mechanisms.
Another clean-slate Internet routing proposal, New Inter-domain Routing Architecture (NIRA) [110], has advocated for adoption of source-based routing schemes for exploring quality end-to-end paths in a scalable fashion by exploiting a hierarchical-address based scheme. Similarly to NIRA, Hybrid Link-state Path-vector Protocol (HLP) [93] exploits hierarchy within cone-like segmentation of Internet topology emerging from customer-provider relationships of ISP peering. Both proposals have targeted to achieve establishment of better paths in terms of QoS and economics, beyond what is offered by BGP, i.e., shortest-path policy routing. The Internet currently evolves into a relatively flat form [34] as opposed to hierarchical model as suggested earlier [33]. With the emergence of various forms of partial-transit and paid-peering relationships, it has become more challenging to classify peering relationships in one of well-known types as provider-customer or peer-to-peer [29]. Nevertheless, NIRA and HLP still stand as very instrumental models where ISPs may form alliances to deliver value-added end-to-end services regardless of the existence of hierarchy. Such alliances are very common in other industries (e.g., airline companies, distribution chains) and are shaped by several common factors, i.e., geographical limits, revenue accumulation, market entry barriers. Telecommunication market is very similar to these industries in several aspects, and size of these alliances would determine the diameter of negotiation capabilities for participating ISPs as we try to investigate in our study.

Both in Pathlet Routing [35] and Contract-Switching [111] frameworks, service providers advertise virtual links between their domain borders. Then, a source ISP can compose end-to-end paths by concatenating these virtual links in a source-routing scheme. These proposals allow multi-hop, multi-metric negotiation of end-to-end path composition process. We focus on the potential benefits of these proposals by
quantitatively analyzing the help of multi-hop negotiations in improving end-to-end path qualities.

4.1.5 Summary and Further Issues

In this work, we analyzed the effectiveness of current local peering policies for improving end-to-end path service quality. Our evaluations on iPlane PoP level Internet topologies with realistic policy setups showed that these local policies are only effective within ISPs’ immediate neighborhood. However, they are not as effective as multi-hop negotiation frameworks proposed by many researchers for exploring better quality alternative paths beyond immediate neighborhood. Multi-hop negotiations are not easily achievable today due to various reasons, e.g., reluctance of major ISPs, rigid SLA structures, spatial and temporal limits of the current Internet market. Our analysis on policy effectiveness shows that multi-hop negotiations are necessary for effectively utilizing opportunities provided by the Internet. However, rich policy settings and mechanisms are required to enable such mechanisms. There are many challenges in the realization of these rich policy settings at the inter-domain routing which call for the further research on several areas such as implementation of policy languages, establishing open standards and developing network protocols which are more inherently policy aware.

4.2 Economic Benefit Analysis from Risk Management Perspective

4.2.1 Bailout Forward Contracts (BFCs)

In general, one can define a contract as an embedding of three major flexibilities in addition to the contracting entities (i.e., buyer and seller): (i) performance component,
(ii) financial component, and (iii) time component. The performance component of a contract can include QoS metrics such as delay or packet loss to be achieved. The financial component of a contract will include various fields to aid entities in making financial decisions related to value and risk tradeoffs involved in engaging in the contract. The basic fields can be various prices, e.g., spot, forward, and usage-based. It is possible to design interesting financial component fields identifying financial security and viability of the contract, e.g., whether or not the contract is insured or has money-back guarantees. The time component can include operational timestamps and be useful for both technical decisions by network protocols and economic decisions by the contracting entities. Example time component fields are the duration the contract will expire in, and the time left for the insured term when the money-back guarantee will expire. Notice that all these three components operate over an aggregation of several packets instead of a single packet. Given the potential scalability issues, this is the right granularity for embedding economic tools into the network protocols instead of finer granularity at the packet level, e.g., per-packet pricing.

In this section, we first formally define a forward contract and a bailout forward contract, and then explain the steps for determining the price of a bailout forward contract. A bailout forward is useful for a provider since it eliminates the risk of demand for bandwidth in the future without imposing a binding obligation to meet the contract if the network cannot support it. A customer of a bailout forward contract locks in the bandwidth required in future, but obtains the bandwidth at a discount. The discount is provided since the customer shares the risk of the scenario that if the network is congested at the future time, the contracted bandwidth may not be delivered due to the bailout clause. The customer may choose not to purchase a forward,
but in that case runs the risk of not being able to obtain the necessary bandwidth at the future time due to congestion or price reasons. Therefore, constructing and offering bailout forwards is beneficial for both providers and customers.

Another key benefit of contracting with bailouts and forwards is to encompass practical issues spanning planning, provisioning and operation of a network into a generic framework. Though these operational issues may take place at different time-scales, our aim is to provide an economic tool that can help ISPs to manage risks or opportunities existing in all layers of ISP management. So, the issue of time-scale will be up to the network manager who will decide when to bailout or forward depending on the sense of risk involved in the network dynamics. Issues like network reliability, over-provisioning, actions taken against failures are all economic decisions as much as technical. Though failures or demand spikes are random events by their nature, the ISP manager has to respond to them as they occur. Further, network management might require proactive actions when opportunities arise without necessarily waiting for failures or demand spikes. Likewise, network security is a risk management or cost optimization problem from planning to its execution [20].

A Spot Contract

A spot contract is the most basic form of contract. The spot prices reflect present utilization of the network and price the contract using either linear or non-linear pricing kernels to promote utilization and cost recovery. The characterization of risks underlying the spot contract prices is key to formulating the pricing framework. Appropriate modeling abstractions are necessary.
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).

Forward contracts, and functionally similar contracts traded in exchanges called ‘futures contracts’, have been common simple derivatives for risk management for eliminating price risk. The underlying asset or instrument to these contracts are varied, ranging from agriculture products, metals, to energy, electricity and weather. Forwards and futures contracts on agricultural products, electricity and weather are closely related with the context of wired bandwidth, where agricultural products are storable, but are likely to be perishable, electricity has limitations on storability, and weather non-storable and perishable. Specific pricing and analysis mechanism must be developed for these cases [16,36,101]. A bandwidth contract of certain duration is storable through the duration of the contract, but decays with the passing time to the point of perishing by the end of the contract term.

A Bailout Forward Contract (BFC)

In the case of a capacititated 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.

Considering multi-provider end-to-end SLA verification problem, both active and passive measurement techniques can be applied as they have been widely used for best-effort services [4]. However, in case of bailout forward contracts, verification protocols which authorize customers and third-party entities to execute temporary verification tasks within provider domain would become instrumental to address potential transparency concerns. Crowd-sourcing mechanisms integrated with multiple customer networks and cloud network paradigm also make it possible for a customer to observe and verify network conditions of a provider domain from multiple perspective. Recent trends have shown great tendency in supporting such capabilities at router level [5].

Further, various bailout “protocols” are possible to implement the bailouts in practice. For example, the provider might decide to bailout and the customer might be given a predetermined amount of time to verify that the bailout conditions really hold or not. Alternatively, the provider might have to get the approval of the customer before bailing out. These protocol possibilities can be negotiated and settled in the bailout clause before the BFC is established.

**Risk Segmentation**

Creation and pricing of a bailout forward contract on a capacitated resource allows for 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.

### 4.2.2 Economic Benefit Analysis of BFCs

For the different contracting schemes with increased computational complexity, although added sophistication in pricing allows more accurate capacity allocation and improves the efficiency of value creation, there is a tradeoff between revenue and computational cost. In order to justify the deployment of more sophisticated contracting, we evaluate and compare the different pricing schemes for their overall economic benefit. In this section, we develop mathematical formulation and derivation of different contracting schemes with different pricing complexity for their benefit analysis. We consider three contracting scenarios (three cases), from the simplest to the most complex in terms of pricing: (i) point-to-anywhere contracts with linear pricing, (ii) point-to-point contracts with non-linear pricing, and (iii) point-to-point, nonlinear with BFCs. In each scenario, we determine the price of contracts, starting with point-to-anywhere spot contracts, followed by point-to-point spot, and finally we model and price bailout forward contracts.

#### Baseline Case 1: Pt-to-anywhere, Linear

In the first scenario, we define the simplest contract scheme. The contracts at each edge node are point-to-anywhere spot contracts, and a flat (linear) pricing scheme is employed, representing the status quo. For a demand profile described below, which is used in all other scenarios, we obtain the optimal flat marginal to determine the point-to-anywhere contract prices. The point-to-anywhere spot prices can be determined for each edge node of a chosen network topology. The pricing scheme is then utilized to compute the revenue generated from this approach during a certain fixed planning
horizon.

In order to implement point-to-anywhere pricing consistent with the point-to-point contracts, demand stream $\mu_{i}^{ij}$ is aggregated for all $j$, defined as aggregate demand $M_i^t$ for each ingress point-to-anywhere node $i$.

$$M_i^t = \sum_j \mu_{i}^{ij}. \quad (4.1)$$

The available capacity for point-to-anywhere traffic is similarly defined as $A_i^t$. The marginal cost of service of all links is similarly aggregated to compute the marginal cost of point-to-anywhere service at node $i$. We choose a demand profile of $N(p, q) = 1 - p - q$, defined as the number or fraction of the customer-base that will buy at least $q$ units at the marginal price $p(q)$ [38]. This demand profile with linear relationship between $p$ and $q$ represents a customer-base that is highly sensitive to price changes. Although other choices of demand-profiles with medium and low sensitivity can be considered [38], we use this sample demand profile to represent a population that sees the best-effort service as a good substitute to the higher quality of service access to bandwidth, i.e. if the price of higher QoS bandwidth goes up, there is a strong tendency of customers to revert to best-effort service. The corresponding demand function is obtained, by integrating the demand-profile over demand $q$, as,

$$D(p) = (1 - p)^2. \quad (4.2)$$

We seek the profit maximizing constant marginal $p^*$, which maximizes the profit function,

$$\text{Profit} = D(p)(p - c_i), \quad (4.3)$$

where $c_i$ is the marginal cost of service at the node $i$. The linear spot price for
point-to-anywhere contract is obtained as,

$$B_i^t = p \frac{M_i^t}{A_i^t}. \quad (4.4)$$

The spot price for a fixed duration of time is used to estimate the aggregate revenue generated from all the point-to-anywhere traffic at node $i$.

**Baseline Case 2: Pt-to-pt, non-linear**

Moving one step up from Baseline Case 1, in this case the provider offers non-linear pricing based point-to-point spot contracts. The comparison of this case with Baseline 1 allows measuring the benefit of added complexity of non-linear pricing of g2g (point-to-point) spot contract. Adding complexity means that now for every g2g link, we conduct more effortful computation in terms of demand modeling and capacity estimation, irrespective of using linear or non-linear pricing.

In a multiple g2g contracts abstraction of the network, we modeled the time-dependent demand for spot contract on each contract link by $\mu_{ij}^t$, and the available capacity on each contract link is modeled by $A_{ij}^t$, defined previously. In non-linear pricing schemes, the demand characteristics of a population or customer-base are often described by a demand profile. The demand profile $N(p(q), q)$ for a commodity is defined as the number or fraction of the customer-base that will buy at least $q$ units at the marginal price $p(q)$. We apply the Ramsey pricing model [102, 103] to determine the optimal price schedule. The guiding principle of the Ramsey pricing model is to develop tariffs that maximize an aggregate of customers’ benefits, subject to the constraint that the provider’s revenues at least recover its total costs, fixed and variable. With our choice of demand profile, $N(p(q), q) = 1 - p(q) - q$, the optimal
price schedule \( p^*(q) \) in the non-linear pricing methodology by Ramsey Rule is:

\[
p^*(\frac{\mu_{ij}^{ij}}{A_{ij}^{ij}}) = \frac{c_{ij} + (1 - \frac{\mu_{ij}^{ij}}{A_{ij}^{ij}}) \times \alpha}{1 + \alpha}.
\] (4.5)

Parameters \( c_{ij} \) and \( \alpha \) are the marginal cost of service for link \( ij \) and the Ramsey number, respectively. Ramsey number captures the extent of monopoly power the provider can exert, where \( \alpha = 1 \) corresponds to a profit-maximizing monopolistic setting and \( \alpha = 0 \) is the perfect competition. We will work with an \( \alpha \) lying in the interval \((0, 1)\) representing an oligopolistic competition. Price of the spot contract is an integral of optimal price schedule:

\[
S_{ij}^{ij} = P(\frac{\mu_{ij}^{ij}}{A_{ij}^{ij}}) = \int_{0}^{\frac{\mu_{ij}^{ij}}{A_{ij}^{ij}}} p^*(q) dq, \quad (4.6)
\]

where integral is taken up to the demanded bandwidth expressed as a fraction of the available bandwidth, \( \frac{\mu_{ij}^{ij}}{A_{ij}^{ij}} \). This definition acknowledges the capacitated nature of the resources as well as encourages high utilization.

Using the g2g spot prices, \( S_{ij}^{ij} \), for the fixed duration of time, estimates of aggregate revenue are generated from all the point-to-point spot contracts for comparison with Baseline Case 1.

**Bailout Forward Contract (BFC): Pt-to-pt, non-linear w/ Bailout Forward**

For a set of g2g spot contracts of the provider’s network, the provider can define and price bailout forward contracts for a chosen set of maturities during the fixed duration of time. It is reasonable to assume that when the provider offers BFCs, a fraction of future demand for spot g2g contracts migrates to g2g BFCs. Under this assumption, the total revenue from g2g spot contracts and forwards is evaluated for the planning period.
To define the bailout feature of the BFC, for available capacity, $A_{ij}^t$, is used to determine the bailout clause and price the BFC on each g2g link of the network.

Following the steps described for solving $F_{ij}$, we obtain

$$F_{ij} = \frac{1}{P(A_{ij}^T > T A_{ij}^T)} E[S_{ij}^T I_{(T A_{ij}^T > T A_{ij}^T)}],$$

(4.7)

where $S_{ij}^T$ evolves in the risk-neutral world.

Based on the fraction of the future demand for spot service that migrates to the set of BFCs offered, we obtain the aggregate revenue generated from all the point-to-point spot and forward contracts in the fixed duration of time.

**Benefit Analysis and Performance Evaluation of BFCs**

With our pricing framework formulated earlier in this section, we now conduct a series of experiments for benefit analysis and performance evaluation. In these experiments, we examine performance of three contracting schemes with increasing levels of complexity on realistic ISP network topologies. The benefit analysis addresses the following important questions:

- **Complexity Trade-off**: How much complexity can be beneficially introduced into the financial component of our contracting schemes?

- **Revenue Comparison**: How does the total revenue gained in the three cases compare?

- **Assessing Value of Information**: What is the value of improved predictability of future demand from the perspective of the provider?
Revenue Comparison and Complexity Tradeoff

We begin our analysis with focusing on the linear pricing scheme as in Baseline Case 1 (henceforth BC1), which is implemented for each node of the Abovenet and Exodus topologies. We identify a period of 7 days to track the revenue comparisons between the three scenarios. Since decision of price complexity needs to be addressed for each link emanating from a node, for illustrative purposes, we track a couple of nodes here. Results obtained for single nodes can be aggregated to give the overall network summary.

We next implement the Baseline Case 2 (henceforth BC2) pricing for each node. In order to address the first research question of complexity, we need to assess the value of treating each point-to-point contract emanating from a node distinctly. In the top panel of Figure 4.11, we plot 95% confidence level of revenue obtained from each of the links emanating from two arbitrarily picked nodes in the Abovenet topology. The level of revenue obtained from a set of links is indistinguishable, which merits that they be merged for pricing purposes. We make six groups of links, each group is priced separately, while each link in the group inherits the pricing for the group. This is the reduced Baseline Case 2 (henceforth Reduced BC2). The revenue level of each link is plotted by its group in the bottom panel of Figure 4.11.

Revenue from BC2 is clearly expected to be higher than BC1, but we need to investigate the impact of reduction in complexity going from BC2 to Reduced BC2. Histograms for total revenue for the 7 day period are plotted in Figure 4.12. After a significant reduction in number of distinct prices being determined in Reduced BC2, the total revenue is much more favorable in Reduced BC2 than in BC1.

Based on the Reduced BC2, we can now introduce the temporal innovation for bailout forward contracts. We define and price the BFCs based on the Reduced BC2
Figure 4.11: ABOVENET: Top panel: 95% Confidence Intervals for Mean Revenue for each link of Node 1 and Node 5 in BC2. Bottom panel: Mean Revenue level for each group of links of Node 1 and Node 5 in Reduced BC2.

Figure 4.12: ABOVENET: Histograms of 7 Day Total Revenue from Baseline Case 1, Baseline Case 2, and Reduced Baseline Case 2 for Nodes 1 and 5.
for the distinctly priced group of links. Figure 4.13 plots the 95% confidence interval of mean revenue obtained from the BFCs over the 7 day period, for both Abovenet and Exodus topologies. Here for the Exodus topology, it is possible to merge some of the BFCs due to the almost indistinguishable revenue they generate. The level of mean revenue generated after the merging is shown in the left panel of Figure 4.13 (four groups of BFCs were made). This is the Reduced BFC implementation for the nodes.

Figure 4.13: Left panel: ABOVENET: 95% Confidence Intervals for Mean Revenue in BFC, and Mean Revenue level after grouping in Reduced BFC (Node 1). Right panel: EXODUS: 95% Confidence Intervals for Mean Revenue in BFC, and Mean Revenue level after grouping in Reduced BFC (Node 2).
A BFC sells bandwidth at a discount, but locks into future deterministic revenue. We now study the level of reduction in revenue from Reduced BFC when compared to Reduced BC2. While Reduced BFC may be inferior than Reduced BC2 in mean revenue terms, we also want to see how much Reduced BFC is able to exceed the baseline setting of BC1. The total revenue histograms in Figure 4.14 are revealing. Reduced BFC significantly dominates the BC1 scenario in terms of total revenue generated.

![Histograms of 7 Day Total Revenue](image)

Figure 4.14: ABOVENET: Histograms of 7 Day Total Revenue for Baseline Case 1, Reduced Baseline Case 2, and Reduced BFC with demand conversion at 40% (Node 1)

**Value of Information**

A provider can sell all its bandwidth in forward contracts, thus lock into deterministic future revenue, thereby possessing perfect information for future demand for bandwidth. Alternately, the provider may not sell any bandwidth in forward contracts, thus obtaining higher revenues from spot contracts, but with higher variability in total revenue. We now analyze the impact of points in the middle of the two extremes
of all or no forward contracts. For all the BFC results displayed so far, the provider lets 40% of its demand for spot contracts migrate or convert to forward contracts. In Figure 4.15, we vary this conversion rate and plot the histogram of revenue from Reduced BFC with different conversion rates of demand from spot to forwards. These are compared with the Reduced BC2 (top-left). It is clear that as the demand conversion rate increases, not only does the total revenue decrease on average, the spread of revenue decreases as well. In other words, the provider is trading-off the mean revenue for the variability or risk in the revenue. The cost of lower variability in revenue is paid in terms of reduction in mean revenue – this is the standard risk-return trade-off.

![Histograms for 7 Day Total Revenue from Reduced BC2](top-left) and Reduced BFC with different demand conversion rate (CR = 20% (top-right), 30% (bottom-left) and 50% (bottom-right)) (Node 1)

We further highlight this trade-off in Figure 4.16. For a range of demand conversion rates between spot and forward contracts, we plot the standard deviation of total revenue versus the confidence interval of mean total revenue. Therefore,
for each standard deviation level, there is an associated demand conversion rate, and the two points in the plot indicate the lower and upper limits of the confidence interval of the corresponding mean revenue. With decreasing conversion rate, hence increasing standard deviation (or risk) of revenue, the mean revenue also increases. For reducing the risk in revenues or obtaining better predictions of future demand for bandwidth, the provider has to give up some of its revenue on average. How much return a provider will be willing to give up for reduction in the risk depends on its risk-aversion. A very risk-averse provider will traverse to the left end of the curve, giving up substantial mean revenue for a significant reduction in risk. This will correspond to selling a high fraction of bandwidth in forward contracts. On the other hand, a less risk-averse provider may function at the right end of the curve, selling only a small fraction of its bandwidth in forward contracts. This risk-return trade-off curve provides a clear view for assessing the value of information for a provider, and determining the optimal mode of operation depending on the provider’s risk-preference.
Figure 4.16: ABOVENET: 95% Confidence Intervals for Mean Revenue for Reduced BFC plotted by the Standard Deviation of Revenue for changing demand conversion rates (CR = 50%, 40%, 30% and 20%) (Node 1)
Chapter 5

Cloud-Assisted Routing

Concerns over the scalability of routing has increased recently. [24, 75, 82, 106] Currently, there are more than 30,000 service providers advertising more than 350,000 different IP prefixes [7, 8]. Today, a typical edge router receives hundreds of updates per second on how to route traffic to these IP prefixes. Sub-linear growth for routing table sizes is considered impossible and a linear growth is the best possible as long as the addressing is performed independent of the underlying topology [19, 60]. Recent analysis of inter-domain routing, i.e., BGP, also points to similar results [48]. Further, BGP churn can grow prohibitively if topology growth and update dampening practices are not performed carefully. [28] These concerns become more serious as the Internet is becoming less hierarchical and more strongly connected [34], putting more burden on the core routers.

In parallel to this growing routing complexity, multi-homing and peering practices as well as the demand on more routing flexibility (e.g., source routing, multi-path routing, QoS routing) have been contributing to this routing complexity issues the Internet faces. Operators rightfully expect more flexibility and programmability in router configurations, which further challenges router architectures by inclusion of software-based designs [26] and virtualization of routing as a service [30, 62].
As the complexity on the routers increased, the cost of a router became non-trivial. The cost of routing unit traffic has not been reducing at a pace similar to the performance improvement of computing capabilities of a router [82]. Given the trends on the state and packet processing capacities expected from a BGP router, the cost of a router that can perform the basic routing functions at the Internet core is unlikely to reduce. These trends clearly point to the urgent need for techniques and architectural approaches reducing or offloading complexities on the routers. In this proposal, we aim to mitigate the increasing routing complexity to the cloud and to seek an answer to the following question: “Can techniques leveraging the memory and computation resources of the cloud remedy the routing scalability issues?”

To address the alarming increase in routing complexity, particularly at the inter-domain level, we introduce a new architectural approach, Cloud-Assisted Routing (CAR). The key contribution of our work is to outline a framework on how to integrate cloud computing with routers and define operational regions where such integration is beneficial. A differentiating property of CAR is its aim to find a middle ground between a pure local approach that targets to scale router performance (e.g., RouteBricks [26]) and a completely cloud-based approach for seamless and highly flexible routing services (e.g., CSR [9]).

Though moving routing functionalities to the cloud is an inevitable trend [61, 64, 87], delegating all routing functions to the cloud has credible risks in exposing critical routing services to potential cascading failures [32]. We believe that hybrid approaches (see Figure 5.1) that maintain high priority tasks at the router and employ an adaptive cloud-router integration framework have more likelihood of addressing future routing scalability and flexibility needs. In that sense, CAR follows an opportunistic model, where routers exploit cloud resources whenever they are
available and beneficial. Such hybrid approach with partial dependence on the cloud resources will allow routers to quickly retake responsibility of delegated tasks in case of cloud-related connectivity or performance problems.

We propose to apply the CAR approach to two well-known routing problems: (1) reduction of peak CPU utilizations at BGP routers, and (2) reduction of forwarding table (RIB and FIB) size that has to be stored at router memory. The rest of the proposal is organized as follows: In Section 5.1, we explain CAR’s architectural view, motivation and principles; and make a detailed discussion of its key tradeoffs. We also discuss where routing functions are placed according to CAR, and the requirements for such placement along with its challenges.

5.1 Cloud-Assisted Routing

5.1.1 CAR Architectural View and Principles

A generic view of our CAR architecture includes a legacy hardware router with partial routing functions and a software router with full routing functions. These two
components establish a hybrid “CAR Router” as illustrated in Figure 5.2. In a way similar to how virtual memory systems use secondary storage to implement the full functionality of the memory, CAR uses the cloud to implement the full functionality of the Router X. Again, similar to the virtual memory systems, Router X will be mostly ‘active’ while Proxy Router X will be mostly ‘passive’.

We envision that a software Proxy Router X in the cloud will hold the full routing and forwarding tables; and will be the default point of service for data and control plane functions that cannot be handled at the hardware Router X. We anticipate that some of the control plane operations such as on-demand route computations due to failures will still be triggered by the hardware Router X. However, CAR will host heavy routing optimizations or periodic protocol updates at the Proxy Router X.

Control Plane

CAR’s approach to scaling computational complexity of routing is to delegate control plane functions to the cloud to the extent possible. This approach has the potential
to scale routing via the cloud’s speed up by parallelism [68]. Though earlier proposals aimed to exploit parallelism in routing by modularizing a router into many parallel working nodes [26], cloud computing offers extensible resources as requested beyond what can be offered locally. Further, many routing related problems only require on-demand, large-scale computations (e.g., traffic engineering based on maximum link utilization, calculating fast reroutes and backup paths) which naturally fit to the CAR’s approach of having an active hardware router.

Data Plane

Currently, data plane is under a heavy strain of high demand on increasing data rates and more flexible management [66]. Current commercial routers (e.g., Juniper T-series or Cisco CRS line [1]) provide scalable data plane with highly optimized packet forwarding circuits. These forwarding circuitry basically first resolves the destination address based on IP header fields, labels of different tunneling protocols or multiple packet fields according to a flow description. Then, a lookup operation is made to figure out which outgoing interface will be used to forward the packet. All these lookup and forwarding operations are hardcoded into extremely fast ASIC circuits and forwarding tables are kept in customized TCAM memories.

CAR’s approach is to calculate the full forwarding tables in the Proxy Router X at the cloud, and then populate the partial forwarding tables on the hardware Router X. These forwarding states can include IP lookup table, MPLS labels, flow descriptor tables for software processing or tunneling information on various layers (VPLS, GRE, IP-to-IP tunnels). There have been many proposals which suggest more flexible and programmable packet classification and forwarding function definitions at data plane [11, 17, 73]. In our CAR architecture, packet classification, flow descriptions,
corresponding forwarding actions (e.g., traffic shaping, DiffServ mechanisms, packet filtering, provisioning) are all defined as “movable states” computed by control plane. The actual Router X is responsible for mapping of these ghost configurations to the lower level networking resources, which can be physical interface addresses, layer 2 configurations, GRE tunnels or other virtualized resources. In brief, CAR’s data plane consists of virtualized low layers, movable upper layer states, and forwarding components used to compose both hardware and software level customizable data plane functions.

**Principles**

CAR’s key goal is to “mitigate the increasing routing complexity to the cloud.” We suggest that, following the Amdahl’s Law, CAR treats the router hardware as a precious resource and thus should focus on the most frequent or important routing functions in the router and offload the rest to the cloud. Particularly, the following two principles should be followed when applying CAR to a routing problem:

**CPU Principle:** *Keep Control Plane Closer to the Cloud.* CAR designer should offload heavy computations to the cloud as much as possible. Example of such heavy but not-so-urgent control plane computations include BGP table exchanges, full-fledged shortest-path calculations, and various traffic engineering optimizations.

**Memory Principle:** *Keep Data Plane Closer to the Router.* CAR designer should keep the packet forwarding operations in the router to the extent possible. An example conformation to the memory principle is to handle most of the forwarding lookups by maintaining a copy of heavily used prefixes at the router memory and delegate the rest of the lookups to the cloud where the complete set of prefixes is held.
5.1.2 CAR Motivations

Cloud for Routing. The idea of using nearby computing resources for improving a router’s performance is not new, though it was considered as a pure software router approach [26]. As the cloud is getting “closer” to router hardware via availability more sites providing cloud computing services, it became possible to make pragmatic comparisons among cloud providers and select the best one fitting to one’s particular needs [64]. Recent measurements [64] show that the latency to the closest cloud provider and response time on various types of computation and storage tasks can be at sub-seconds levels with strong statistical reliability. In this context, we believe that using cloud services to relieve routers’ complex duties presents a great opportunity. Further, such integration of cloud services to router platforms will enable intra-domain and inter-domain routing optimizations by leveraging the “central” role of the cloud.

Hybrid Routers: Software and Hardware. As the Internet core is becoming more datacenter-oriented, the tendency of moving more distributed functions like routing has increased, e.g., NEBULA [2]. This tendency is mainly because of the significant potential for optimizations when things are done in a centralized manner. However, the nature of functions like routing requires distributed operation which hinders the optimality. Solutions performing as a hybrid in between these two well-justified extremes will always be needed. CAR as a framework allows exploration of such hybrids between software-based centralized routing optimizations at the cloud and the traditional hardware-based decentralized routing.

Locality. Several studies observed temporal (bursts of packets in the same flows) and spatial (few popular destinations) locality in data packet traffic [59]. This means that even though most of the destinations will be looked up very infrequently, they will
keep occupying the routing table. Likewise, even though most of the updates and routing table entries will be needed very infrequently, router CPU(s) will be consumed for those updates and entries. A key motivation for CAR is to leverage these locality patterns and delegate the less used majority to the cloud while keeping the more used minority at the router.

5.1.3 CAR Challenges and Tradeoffs

To-Delegate or Not-To-Delegate. Due to the extra delay coming from delegation to the cloud, CAR designer’s key metric to decide how to split functions to be delegated to the cloud is the cloud-router delay, $t_{CR}$. Intuitively, if $t_{CR}$ is too high, the designer should not delegate to the cloud and keep more routing functions at the router to achieve higher overall efficiency. But, for a limited hardware router, other factors such as rates of traffic flows and router’s buffer size will play roles in identifying which flows to delegate. In general, the flows delegated to the cloud will receive a ‘routing service’ with larger delays and smaller loss; and thus, keeping a fair treatment of data flows is a challenge CAR faces. CAR’s contribution here is a framework for exploring efficiency-fairness tradeoff, potentially on a flow-based manner.

Adaptive Tuning. The key indicator for fruitfulness of CAR is whether it is possible to achieve a similar performance with smaller router hardware resources like memory. Just like the virtual memory does not pay off if there is no locality, CAR will have to actively leverage the locality patterns in traffic to yield benefits over the existing router designs. This requires adaptive tuning of caching and delegation of router’s functions for different traffic patterns and situations. The benefit of CAR is going to be highly dependent on the effectiveness of this adaptive tuning. We anticipate that such adaptive tuning will not be hard to do given the locality and regularity of the
traffic patterns arriving at a router.

5.2 Proposed Tasks

Based on the CAR framework, we propose to build two prototypes where some of the well-known routing tasks (one for CPU-heavy tasks and another for memory-heavy tasks) can be offloaded to the cloud.

5.2.1 CAR I: Peak CPU Utilization for BGP Peering

Recent studies on performance of routing have shown BGP protocol consumes most of the CPU resources during full table exchanges with peer routers during BGP peer establishment or BGP peer reset events. Although there are many mechanisms for mitigating the cost of peer re-establishment (e.g., prefix filtering or preventing table exchanges after session reset), with the growing size of BGP tables and increased need on dynamic traffic engineering, CPU resource consumption of BGP protocol will seem to be a challenging issue for router performance.

During full table exchange, BGP route selection algorithm, ISP policy (of preference, filtering) is applied on each path learnt from peer for each prefix. Considering the highly-selective process and steady-state of router, it can be safely assumed that most of the full table exchanges only result in a few significant changes on forwarding state and consequent BGP updates. According to that assumption, rest of the learnt paths will be included in RIB and only used as backup paths if the primary (most preferable) path will fail. However, BGP route selection algorithm should be applied on all the paths learnt from peers whether they will selected or not as primary paths at the end of the process.

Building upon this observation, we propose a two phase peer establishment
Figure 5.3: CAR: Peering Establishment Scenario

mechanism which exploits computational and storage resources of cloud architecture. As seen in 5.3, prior to peer establishment, during capability exchange phase, routers will inform if they are cloud-assisted routing capable by exchanging their cloud router proxy address and secure communication parameters. Second step is to authorization of peer establishment between cloud router proxies. Cloud-router proxies which assumingly keep exact copies of the routing and forwarding state of actual routers, will carry out tasks related to full table exchange and BGP route selection processes as if they are regular routers. Upon completion of these processes, cloud-proxy routers classify the changes on routing and forwarding states as priority and non-priority. So, if the preference of an AS-PATH will lead to forwarding path changes and BGP updates, this entry will be marked as priority and immediately shared with actual router. Then, finally only entries with marked prefixes will be exchanged between actual routers, filtering out all other entries for the first phase by using already available mechanisms described by RFC 5292. So, the actual router only has to consume its computational resources on significant updates. Peak usage of computational resources on BGP will be spread over time as changes on secondary paths will be suppressed.
5.2.2 CAR II: BGP Forwarding Table Size

A well-known issue with core BGP routers is their forwarding table (FIB) and routing table (RIB) sizes. We propose to apply the CAR concept to reducing the FIB size. The idea is to store only partial FIB at the router hardware and delegate packets to the cloud if a miss occurs during the lookup at the partial FIB. The proxy at the cloud will store the full FIB, and thus will be able to handle any misses at the router hardware. As shown in Figure 5.4, we plan to implement this relationship between the hardware router and the proxy at the cloud via VLAN establishment. We will explore other possibilities as well, such as tunnels or dedicated TCP sessions.

We plan to handle FIB lookups in a hierarchical memory organization, as shown in Figure 5.4. For instance, some packets will be handled completely via hardware lookups (e.g., packets destined to 8.8/16 in the figure), some via software lookups at the router (e.g., packets destined to 72.24.10/24 in the figure), and some via lookups at the cloud (e.g., packets destined to 72.36.10/24 in the figure).

Similar to traditional cache organizations, the lookup will be delegated to one level up in the hierarchy if a miss occurs. However, unlike the traditional cache
systems, the level below will store entries corresponding to an aggregate of multiple entries at the level above if the real entries are to be stored at the level up. For example, if we decide to place an entry under the prefix 72.36.10/24 to the cloud, then all other other entries under that prefix will be moved to the cloud as well. Though holes could be punched and some “important” entries could be placed at the router, motivation for such complication, in our opinion, is not that high – particularly when the delay to the closest cloud can be pretty short in today’s Internet.

In general, the placement of prefix entries to the different levels of this CAR framework is not an easy task as explored previously in different contexts [13, 59]. It involves several dynamic parameters such as lookup frequency of prefixes and importance of prefixes due to their contractual value. The positive factor is that high locality patterns exist in these parameters. Those prefixes that are delegated to the cloud will suffer from additional delays, and a key goal will be to establish an acceptable fairness across prefixes. We will explore heuristics that can adapt to these dynamics and reorganize the FIB placement in CAR so that the locality can be leveraged to the extent possible while keeping an overall fairness across flows/prefixes.

5.3 Previous Research

The Internet’s routing has grown to be a complex, customizable service which cannot be left to the routers alone [30]. The concept of “Routing As a Service” (RaaS) [62] implies the separation of control and data planes where routing decisions are made and executed. Such a separation can be very beneficial where control plane tasks are delegated to “clouds” which offer as vast computational power, storage and parallelism required by the enlarging and diversifying routing problem. Path calculation with respect to multiple distance metrics (e.g., bandwidth, latency, loss rate, price) on
wide-range of possible IPv6 address space would be challenging on existing routers with their limited capacities [72]. Parallel router architectures [26,41,63,108], network processors [104,107] or GPU-empowered routers [42] can mitigate this complexity temporarily. However, it is still a question whether these approaches can prepare routers for the next billion of Internet users with more challenging application traffic requirements [6]. Yet, *cloud computing offers easily extendable capacity that may address these challenges.*

Moreover, with the high computational power and parallel execution capabilities made available to router designer, it becomes possible to consider complex aspects of reliability [100] and economics [111] as routing functions. Even custom data forwarding optimizations can be considered such as lookup performance improvements, aggregation of routing table entries and forwarding rules for TCAMs [97]. These operations are inherently computation-heavy tasks which are usually not being computed in an online manner. Delegation of these computation- and memory-intensive operations to the cloud is the key inspiration of our work.

Routing tasks consume much of the resources on current routers [104,106]. Classification of these tasks carried out by a router as "delegatable" or "in-place" will be a first step towards delegating some of them to a cloud-based control plane. Then, released resources by offloaded delegatable tasks can be reclaimed by data plane to offer enhanced in-place services. Location-based characteristics of these in-place (or in-situ) services require them to be executed on routers instead of a remote location (e.g., cloud). Mobility, security, provisioning, traffic management, virtualization and monitoring can be given as examples of such in-situ services. Along with the delegation of routing tasks and simplified architecture of routers, these network entities can be better designed to support virtualization and programmability as other
researchers have also suggested [5, 11, 17, 18, 73].

As an approach to connect cloud computing resources with enterprise networks, CloudNet [105] enables users to control network resources in the cloud as well as computational and storage services. CloudNet offers integration of cloud resources with enterprise networks through Virtual Private Networks (VPN) and partitioning of cloud resources into multiple Virtual Private Clouds (VPCs). We believe that such integration of cloud resources into enterprise networks via greater network control can be utilized in the realization of data and control plane separation for Internet Service Providers (ISPs). Cloud-enabled ISPs may delegate heavy-computational tasks of routing control plane to the cloud. These cloud-enabled ISPs can achieve true virtualization in their backbones and offer virtualized forwarding plane infrastructure as cloud service platforms to many parties including content providers, content distribution networks and other cloud-enabled ISPs.
Chapter 6
Conclusions and Future Work

6.1 Conclusions

Current Internet Architecture only allows local negotiation of a single end-to-end path between a source and destinations on the Internet over a single performance metric. However, advanced value added-services cannot be realized by using these limited negotiation capabilities. In this proposal, we propose Contract Routing Architecture that employs contract definitions, which are the combined representation of both topology, e.g., link, address, and economic value, e.g., expected utility and service price, for routing. Such an approach enhances the representation of network domain capabilities and service definitions for ISPs as well as Economic value expression in provided and demanded services.

We show that multi-metric, multi-lateral, end-to-end path negotiations can be achieved in a scalable manner within Contract Routing Architecture. To achieve scalability, first we investigate path exploration models that exploit finer-time granularities in SLAs, query-based, indeterministic information dissemination methods, and structural characteristics of Internet topology which inherently provides rich diversity of end-to-end paths.
Moreover, we show that multi-lateral, multi-metric negotiations can achieve 50-90\% performance gains in terms of multiple network performance metrics, e.g., latency and packet loss rate. Also, we show that many flexible economic service level agreement models can be implemented on top of contract definitions so as to amend current broken revenue model for Internet service providers.

6.2 Future Work

As future work, we plan to elaborate our studies with following items:

- We would like to put Path-Vector and Link-State Contract Routing protocols’ information dissemination methods into theoretical framework. We are currently working on epidemic-modeling of Path-Vector Contract Routing. Also, another interesting extension to information dissemination model of PVCR is to explore more directed and structured approaches.

- We also want to investigate network-formation models to formulate alliance aspects of both PVCR and LSCR where a set of ISPs act as an alliance to negotiate end-to-end paths as well as offering integrated services to other alliances. Alliance model would also serve as an aggregation method within Contract Routing Architecture as a way of achieving scalability.
Bibliography


