Maximum Throughput Routing of Traffic in the Hose Model

M. Kodialam T. V. Lakshman Sudipta Sengupta
Bell Laboratories, Lucent Technologies, Holmdel, NJ, USA
Massachusetts Institute of Technology, Cambridge, MA, USA

Abstract— **Routing traffic subject to hose model constraints [5] has been of much recent research interest [10], [6]. Two-phase routing has been proposed [9], [18] as a mechanism for routing traffic in the hose model. It has desirable properties in being able to statically preconfigure the transport network and in being able to handle constraints imposed by specialized service overlays.**

In this paper, we investigate whether the desirable properties of two-phase routing come with any resource overhead compared to (i) direct source-destination path routing, and (ii) optimal scheme among the class of all schemes that are allowed to even make the routing dynamically dependent on the traffic matrix. In the pursuit of this endeavor, we achieve several milestones. First, we develop the first polynomial size linear programming (LP) formulation for maximum throughput routing of hose traffic along direct source-destination paths. Second, we develop the first polynomial size LP formulation for maximum throughput two-phase routing of hose traffic for a generalized version of the scheme proposed in [9]. Third, we prove that the throughput of two-phase routing is at least $1/2$ that of the optimal scheme. Using the polynomial size LP formulations developed, we compare the throughput of two-phase routing with that of direct source-destination path routing and optimal scheme on actual Internet Service Provider topologies collected for the Rocketfuel project. Quite surprisingly, the throughput of two-phase routing matches that of direct source-destination path routing and is within 6% of the optimal scheme on all evaluated topologies. We conclude that two-phase routing achieves its robustness to traffic variation without imposing any appreciable additional resource requirements over previous approaches.

I. Introduction

With the rapid rise in new Internet-based applications, such as peer-to-peer and voice-over-IP, it has become increasingly important to accommodate widely varying traffic patterns in networks. This requires Internet Service Providers (ISPs) to accurately monitor traffic and to deploy mechanisms for adapting network routing to changing traffic patterns. This dynamic adaptation increases network operations complexity. To avoid this complexity, service providers would like to provision their networks such that the provisioning is robust to large changes in the traffic pattern. This has led to interest in the *hose traffic model* [5] where the assumption is that we have knowledge regarding the maximum traffic entering and leaving the network at each node but we do not have knowledge of the actual traffic matrix itself. Several algorithms for routing traffic in the hose model have recently been proposed. These schemes [5], [10], [6] route traffic *directly from source to destination along fixed paths*.

A recently proposed approach is *two-phase routing* [9], [18]. Here, in the first phase incoming traffic is sent from the source to a set of intermediate nodes in predetermined proportions and then, in the second phase, from the intermediate nodes to the final destination. The proportion of traffic that is distributed to each intermediate node in the first phase can depend on the intermediate nodes as proposed in [9].

The two-phase routing scheme is flexible in being able to handle wide traffic variations and in being useful for various networking applications such as service overlays with bandwidth guarantees, virtual private networks, routing through middleboxes for security, and IP-over-Optical networks with a statically configured transport layer.

For the IP-over-optical network application, the requirement of static provisioning at the optical layer necessitates that not only the paths but also their associated bandwidths do not change with shifts in traffic. Two-phase routing meets this requirement while direct source-destination path routing does not. An important innovation of two-phase routing scheme is the handling of traffic variability in a capacity efficient manner through static preconfiguration of the network and without requiring either (i) measurement of traffic in real-time or (ii) reconfiguration of the network in response to changes in it. This is explained in Section III-B.

In this paper, we investigate whether the desirable properties of two-phase routing come with any resource overhead compared to (i) direct source-destination path routing, and (ii) optimal scheme among the class of all schemes that are allowed to even make the routing dynamically dependent on the traffic matrix. In the pursuit of this endeavor, we achieve several milestones. First, we develop the first polynomial size linear programming (LP) formulation for maximum throughput routing of hose traffic along direct source-destination paths. Second, we develop the first polynomial size LP formulation for maximum throughput two-phase routing of hose traffic for a generalized version of the scheme proposed in [9]. Third, we prove that the throughput of two-phase routing is at least $1/2$ that of the optimal scheme.

Using the polynomial size LP formulations developed, we compare the throughput of two-phase routing with that of direct source-destination path routing on actual ISP topologies collected for the Rocketfuel project [16]. Quite surprisingly, the throughput of two-phase routing matches that of direct source-destination path routing and is within 5% of the optimal scheme on all evaluated topologies. We view our work as an effort in dispelling any concerns that two-phase routing achieves its robustness to traffic variation through substantial over-provisioning of capacity. We believe that establishment of

1-4244-0222-0/06/$20.00 (c)2006 IEEE

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the Proceedings IEEE Infocom.

the capacity efficiency of two-phase routing through rigorous investigation, as attempted in this paper, will influence and shape the mindset of ISPs in considering its deployment in their networks.

In this paper, we use the reciprocal of the maximum link utilization, also known as throughput, as the optimization metric. This is a commonly used metric in the literature since it is directly related to it is directly related to other metrics like link congestion.

The paper is structured as follows. In Section II, we first survey related work for direct source-destination path routing and then develop the polynomial size LP for maximizing throughput. In Section III, we begin with an overview of the two-phase routing scheme and bring out its benefits over direct source-destination path routing. We then discuss a generalization of the scheme and develop the polynomial size LP for maximizing throughput. Finally, we establish the 2-optimality bound for two-phase routing. In Section IV, we discuss a method for upper bounding the throughput of the optimal scheme. In Section V, we compare the throughput performance of two-phase routing with that of direct source-destination path routing and the optimal scheme on actual ISP topologies collected for the Rocketfuel project. We conclude and point to future work in Section VI. We briefly describe some notation and the traffic variation model before proceeding further.

A. Notation

We assume that we are given a network $G = (N, E)$ with node set N and (directed) edge set E where each node in the network can be a source or destination of traffic. Let $|N| = n$ and $|E| = m$. The nodes in N are labeled $\{1, 2, \ldots, n\}$. The sets of incoming and outgoing edges at node i are denoted by $E^-(i)$ and $E^+(i)$ respectively. We let (i, j) represent a directed link in the network from node i to node j. To simplify the notation, we will also refer to a link by e instead of (i, j). The capacity of link (i, j) will be denoted by u_{ij}. The *utilization* of a link is defined as the maximum traffic usage on the link divided by its capacity.

B. Traffic Variation Model

We consider a traffic variation model where the total amount of traffic that enters (leaves) an ingress (egress) node in the network is bounded by the total capacity of all external ingress links at that node. This is known as the *hose model* and was proposed by Fingerhut et al . [7] and subsequently used by Duffield et al. [5] as a method for specifying the bandwidth requirements of a Virtual Private Network (VPN). Note that the hose model naturally accommodates the network's ingress-egress capacity constraints.

We denote the upper bounds on the total amount of traffic entering and leaving the network at node i by R_i and C_i respectively. The point-to-point matrix for the traffic in the network is thus constrained by these ingress-egress link capacity bounds. These constraints are the only known aspects of the traffic to be carried by the network, and knowing these is equivalent to knowing the row and column sum bounds on the traffic matrix. That is, any allowable traffic matrix $T = [t_{ij}]$ for the network must obey

$$\sum_{j \in N, j \neq i} t_{ij} \leq R_i, \qquad \sum_{j \in N, j \neq i} t_{ji} \leq C_i \ \forall \ i \in N$$

For given R_i and C_i values, denote the set of all such matrices that are partially specified by their row and column sums by $\mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$, that is

$$\mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}}) = \{[t_{ij}] \mid \sum_{j \neq i} t_{ij} \leq R_i \text{ and } \sum_{j \neq i} t_{ji} \leq C_i \ \forall \ i\}$$

Note that the traffic distribution T could be any matrix in $\mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$ and could change over time. We will use $\lambda \cdot \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$ to denote the set of all traffic matrices in $\mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$ with their entries multiplied by λ.

II. Direct Source-Destination Routing Along Fixed Paths

Direct routing from source to destination (instead of in two phases) along *fixed* paths for the hose traffic model has been considered by Duffield et al. [5] and Kumar et al. [10]. In order to make throughput comparisons with two-phase routing, we consider the *multi-path* version of direct source-destination routing where the traffic from a source to a destination can be split along multiple paths – both the paths and the ratios in which traffic is split among them is fixed a priori. An instance of this scheme is completely described by specifying how a unit flow is (splittably) routed between each source-destination pair in the network.

In related work, Azar et al. [2] consider direct source-destination path routing and show how to compute *relative guarantees* for routing an arbitrary traffic matrix with respect to the best routing for that matrix. However, they do not provide *absolute bandwidth guarantees* for routing variable traffic under the hose model.

Erlebach and Rüegg [6] consider the problem of minimum cost direct source-destination (multi-)path routing of hose traffic under given link costs (and, link capacities). They give an LP with an infinite number of constraints (and, a polynomial size separation oracle LP) that is suitable for solving using the ellipsoid method [14]. The ellipsoid method is primarily a theoretical tool for proving polynomial-time solvability – its running time is not feasible for practical implementations. The authors also give a cutting-plane heuristic for solving the infinite size LP and obtain reasonable running times for the experiments reported. However, this cutting-plane heuristic can have exponential running times in the worst case.

As one of the key contributions of this paper, we give the first polynomial size LP for maximum throughput multi-path routing of hose traffic under given link capacities. Our technique can be used to obtain a polynomial size LP for the minimum cost version of the problem also, thus improving the result in [6].

A. Throughput Maximization

Given a network with link capacities u_e and constraints R_i, C_i on the ingress-egress traffic, we consider the problem

of direct source-destination path routing so as to maximize the network throughput. The throughput is the maximum multiplier λ such that all matrices in $\lambda \cdot \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$ can be feasibly routed under given link capacities.

We begin with an LP formulation with an infinite number of constraints and a polynomial size separation oracle LP for it, and then combine the two into a polynomial size LP that can be solved in polynomial time using a general linear programming algorithm [14].

LP with Infinite Constraints and Separation Oracle

The fixed path routing for each source-destination pair (i, j) can be specified by a set of *unit flow variables* f_e^{ij}, where f_e^{ij} denotes the fraction of traffic from i to j that traverses link e in the network. Let μ denote the maximum utilization of any link in the network. Maximizing the throughput is equivalent to minimizing the maximum link utilization μ.

$$\text{minimize} \quad \mu$$

subject to

$$\sum_{e \in E^+(k)} f_e^{ij} - \sum_{e \in E^-(k)} f_e^{ij} = \begin{cases} +1 & \text{if } k = i \\ -1 & \text{if } k = j \\ 0 & \text{otherwise} \end{cases} \quad \forall\ i, j, k \in N \quad (1)$$

$$\sum_{i,j \in N} t_{ij} f_e^{ij} \leq \mu u_e \ \forall\ e \in E,\ \forall\ [t_{ij}] \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}}) \quad (2)$$

$$f_e^{ij} \geq 0 \ \forall\ e \in E,\ \forall\ i, j \in N \quad (3)$$

Constraints (1) correspond to routing of unit flows between each source-destination pair for determining the fixed paths. Constraints (2) are the maximum utilization constraints for each link. The quantities t_{ij} in the LHS of (2) are constants and hence the constraints are linear. Note that there is an infinite set of constraints in (2), since there are m constraints for each $[t_{ij}] \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$.

The above LP can be solved in polynomial time by the ellipsoid algorithm [14] provided we can find a polynomial time separation oracle for the constraints (2). Given a set of values for the variables in the above LP, the separation oracle needs to identify at least one constraint that is violated (if any), or indicate otherwise. Clearly, constraint (1) can be verified in polynomial time.

To determine if the constraints in (1) are violated for any link, we need to either identify a link e and a traffic matrix $[t_{ij}] \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$ such that the corresponding constraint is violated, or determine that all such constraints are satisfied. This can be done by verifying that for each link $\ell \in E$, the LP below, with variables $t_{ij}\ \forall\ i, j \in N$, has optimum objective function value at most μ. If not, the traffic matrix $[t_{ij}]$ obtained in the optimal solution of the LP identifies the corresponding violating constraint in (2).

$$\text{maximize} \quad \textstyle\sum_{i,j \in N} f_\ell^{ij} t_{ij} / u_\ell$$

subject to

$$\sum_{j \in N, j \neq i} t_{ij} \leq R_i \ \forall\ i \in N \quad (4)$$

$$\sum_{i \in N, i \neq j} t_{ij} \leq C_j \ \forall\ j \in N \quad (5)$$

$$t_{ij} \geq 0 \ \forall\ i, j \in N \quad (6)$$

As mentioned earlier, the ellipsoid algorithm gives running times that are not feasible for practical implementations. Hence, the motivation for designing a polynomial size LP for the above problem. Such an LP can be directly fed into LP solvers like CPLEX [4] for solution.

Polynomial Size LP

In developing the polynomial size LP, we first take the dual of the separation oracle LP above. For a given link ℓ, the dual LP has non-negative variables $r(i, \ell)$ corresponding to each constraint in (4) and non-negative variables $c(j, \ell)$ corresponding to each constraint in (5).

$$\text{minimize} \quad \textstyle\sum_{i \in N} R_i r(i, \ell) + \sum_{j \in N} C_j c(j, \ell)$$

subject to

$$r(i, \ell) + c(j, \ell) \geq f_\ell^{ij} / u_\ell \ \forall\ i, j \in N \quad (7)$$

$$r(i, \ell), c(i, \ell) \geq 0 \ \forall\ i \in N \quad (8)$$

It follows directly from strong duality of linear programming [14] that for each link $\ell \in E$, the primal (separation oracle) LP has an optimum objective function value of at most μ if and only if the dual LP has a feasible solution with objective function value at most μ.

The requirement that the dual LPs, for all $\ell \in E$, have feasible solutions with objective function value at most μ can be modeled as the following constraint:

$$\sum_{i \in N} R_i r(i, \ell) + \sum_{j \in N} C_j c(j, \ell) \leq \mu \ \forall\ \ell \in E$$

This allows us to remove the infinite set of constraints in (2) and add the above constraint and constraints (7)-(8) from the dual LPs to obtain the following polynomial size LP for our problem:

$$\text{minimize} \quad \mu$$

subject to

$$\sum_{e \in E^+(k)} f_e^{ij} - \sum_{e \in E^-(k)} f_e^{ij} = \begin{cases} +1 & \text{if } k = i \\ -1 & \text{if } k = j \\ 0 & \text{otherwise} \end{cases} \quad \forall\ i, j, k \in N \quad (9)$$

$$r(i, \ell) + c(j, \ell) \geq f_\ell^{ij} / u_\ell \quad \forall \ell \in E,\ \forall\ i, j \in N \quad (10)$$

$$\sum_{i \in N} R_i r(i, \ell) + \sum_{j \in N} C_j c(j, \ell) \leq \mu \ \forall\ \ell \in E \quad (11)$$

$$r(i,\ell), c(i,\ell) \geq 0 \ \forall i \in N, \forall \ell \in E \quad (12)$$
$$f_e^{ij} \geq 0 \ \forall e \in E, \forall i,j \in N (13)$$

This LP has $n^2(n-1)$ constraints in (9), $mn(n-1)$ constraints in (10), m constraints in (11), $2mn$ constraints in (12), and $mn(n-1)$ constraints in (13), for a total of $O(mn^2)$ constraints. The number of variables is $mn(n-1)+2mn+1 = O(mn^2)$.

The technique of using the dual of the separation oracle linear program to replace the corresponding set of infinite (or, exponential) number of constraints in the main linear program has been used earlier in the literature (see, for example, [11]).

III. Two-Phase Routing

We begin with an overview of two-phase routing. We then generalize the traffic split ratios to depend on source and destination nodes also, as proposed in [9]. This is conceivably the most general form of two-phase routing. We develop a polynomial size LP for maximum throughput two-phase routing (with generalized traffic split ratios) of hose traffic under given link capacities. This will serve to compare the resource requirements of two-phase routing with that of direct source-destination path routing of hose traffic.

A. Overview of Two-Phase Routing

In this section, we give an overview of the two-phase routing scheme from [9]. As mentioned earlier, the scheme does not require the network to detect changes in the traffic distribution or reconfigure the network in response to it. The only assumption about the traffic is the limits imposed by the ingress-egress constraints at each node, as outlined in Section I-B.

As is indicative from the name, the routing scheme operates in two phases:

- **Phase 1:** A predetermined fraction α_j of the traffic entering the network at any node is distributed to every node j *independent of the final destination of the traffic*.
- **Phase 2:** As a result of the routing in Phase 1, each node receives traffic destined for different destinations that it routes to their respective destinations in this phase.

This is illustrated in Figure 1. Note that the traffic split ratios $\alpha_1, \alpha_2, \ldots, \alpha_n$ in Phase 1 of the scheme are such that $\sum_{i=1}^{n} \alpha_i = 1$. A simple method of implementing this routing scheme in the network is to form *fixed bandwidth paths between the nodes*. In order to differentiate between the paths carrying Phase 1 and Phase 2 traffic, we will refer to them as Phase 1 and Phase 2 paths respectively. The critical reason the two-phase routing strategy works is that the *bandwidth required for these tunnels depends on the ingress-egress capacities R_i, C_i and the traffic split ratios α_j but not on the (unknown) individual entries in the traffic matrix.* Depending on the underlying routing architecture, the Phase 1 and Phase 2 paths can be implemented as IP tunnels, optical layer circuits, or Label Switched Paths in Multi-Protocol Label Switching (MPLS) [13].

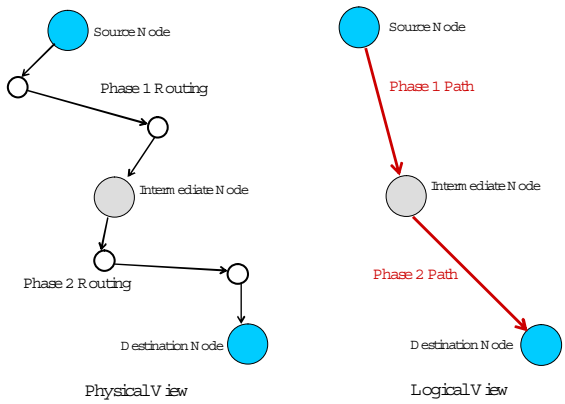

Fig. 1. Two-Phase Routing.

We now derive the bandwidth requirement for the Phase 1 and Phase 2 paths. Consider a node i with maximum incoming traffic R_i. Node i sends $\alpha_j R_i$ amount of this traffic to node j during the first phase for each $j \in N$. Thus, the traffic demand from node i to node j as a result of Phase 1 routing is $\alpha_j R_i$. At the end of Phase 1, node i has received $\alpha_i R_k$ traffic from any other node k. Out of this, the traffic destined for node j is $\alpha_i t_{kj}$ since all traffic is initially split without regard to the final destination. The traffic that needs to be routed from node i to node j during Phase 2 is $\sum_{k \in N} \alpha_i t_{kj} \leq \alpha_i C_j$. Thus, the traffic demand from node i to node j as a result of Phase 2 routing is $\alpha_i C_j$.

Hence, the maximum demand from node i to node j as a result of routing in Phases 1 and 2 is $\alpha_j R_i + \alpha_i C_j$. Note that this does not depend on the matrix $T \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$. The scheme handles variability in traffic matrix $T \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$ by effectively routing the fixed matrix $D = [d_{ij}] = [\alpha_j R_i + \alpha_i C_j]$ that depends only on aggregate ingress-egress capacities and the traffic split ratios $\alpha_1, \alpha_2, \ldots, \alpha_n$, and not on the specific matrix $T \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$. This is what makes the routing scheme oblivious to changes in the traffic distribution.

An instance of the scheme requires specification of the traffic split ratios $\alpha_1, \alpha_2, \ldots, \alpha_n$ and routing of the Phase 1 and Phase 2 paths.

B. Benefits of Two-Phase Routing

In this section, we briefly discuss some properties of two-phase routing that differentiate it from direct source-destination path routing. We consider aspects of two different application scenarios to bring out the benefits of two-phase routing.

Static Optical Layer Provisioning in IP-over-Optical Networks

Core IP networks are often deployed by interconnecting routers over a switched optical backbone. When applied to such networks, direct source-destination path routing routes packets from source to destination along direct paths in the optical layer. Note that even though these paths are fixed a priori and do not depend on the traffic matrix, their *bandwidth requirements change* with variations in the traffic matrix.

Thus, bandwidth needs to be deallocated from some paths and assigned to other paths as the traffic matrix changes. (Alternatively, paths between every source-destination pair can be provisioned a priori to handle the maximum traffic between them, but this leads to gross overprovisioning of capacity, since all source-destination pairs cannot simultaneously reach their peak traffic limit in the hose traffic model.) This necessitates (i) detection of changes in traffic patterns and (ii) *dynamic reconfiguration* of the provisioned optical layer circuits (i.e., change in bandwidth) in response to it. Both (i) and (ii) are difficult functionalities to deploy in current ISP networks.

The (current) traffic matrix is not only difficult to estimate but changes in the same may not be detectable in real time. Direct measurement methods do not scale with network size as the number of entries in a traffic matrix is quadratic in the number of nodes. Moreover, such direct real-time monitoring methods lead to unacceptable degradation in router performance. In reality, only aggregate link traffic counts are available for traffic matrix estimation. SNMP (Simple Network Management Protocol) provides this data via incoming and outgoing byte counts computed per link every 5 minutes. To estimate the traffic matrix from such link traffic measurements, the best techniques today give errors of 20% or more [12].

Moreover, dynamic changes in routing in the network may be difficult or prohibitively expensive from a network operations perspective. In spite of the continuing research on IP-Optical integration, network deployments are far away from utilizing the optical control plane to provide bandwidth provisioning in real-time to the IP layer. The unavailability of network control plane mechanisms for reconfiguring the network in response to and at time-scales of changing traffic amplifies the necessity of *static provisioning at the optical layer* in any scheme that handles traffic variability. Direct source-destination path routing does not meet this requirement.

To illustrate this point, consider the scenario in Figure 2 for direct source-destination routing in IP-over-Optical networks. Here, router A is connected to router C using 3 OC-48 connections and to Router D using 1 OC-12 connection, so as to meet the traffic demand from node A to nodes C and D of 7.5 Gbps and 600 Mbps respectively. Suppose that at a later time, traffic from A to C decreases to 5 Gbps, while traffic from A to D increases to 1200 Mbps. Then, the optical layer must be reconfigured so as to delete one OC-48 connection between A and C and creating a new OC-12 connection between A and D. As such, the *requirement of static provisioning at the optical layer is not met*.

Two-phase routing, as envisaged for IP-over-Optical networks, establishes the fixed bandwidth Phase 1 and Phase 2 paths at the optical layer. Thus, the *optical layer is statically provisioned* and does not need to be reconfigured in response to traffic changes. IP packets are routed end-to-end with *IP layer processing at a single intermediate node only*.

Indirection in Specialized Service Overlay Networks

The Internet Indirection Infrastructure (i3) was proposed in [17] to ease the deployment of services – like mobility, multicast and anycast – on the Internet. i3 provides a rendezvous-based communication abstraction through indirection – sources send packets to a logical identifier, and receivers express interest in packets sent to an identifier. The rendezvous points are provided by i3 servers that forward packets to all receivers that express interest in a particular identifier. The communication between senders and receivers is through these rendezvous points over an overlay network.

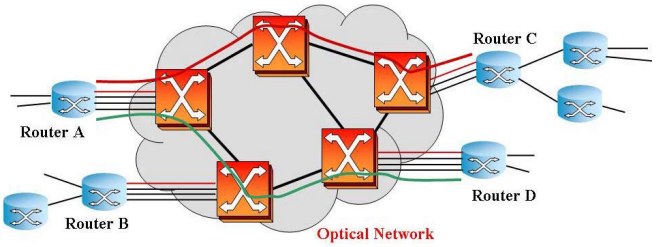

Fig. 2. Routing through direct optical layer circuits in IP-over-Optical networks.

Two-phase routing can be used to provide QoS guarantees for variable traffic and support indirection in intra-ISP deployments of specialized service overlays like i3. (Note that we are not considering Internet-wide deployment here.) The intermediate nodes in the two-phase routing scheme are ideal candidates for locating i3 servers. Because we are considering a network whose topology is known, the two-phase routing scheme can be used to not only pick the i3 server locations (intermediate nodes) but also traffic engineer paths for routing with bandwidth guarantees between sender and receiver through i3 server nodes.

In service overlay models like i3, the *final destination of a packet is not known at the source* but only at the rendezvous nodes. Because the final destination of a packet needs to be known only at the intermediate nodes in two-phase routing, it is well-suited for specialized service overlays as envisaged above. In contrast, for direct source-destination path routing, the source needs to *know the destination of a packet* for routing it, thus rendering it unsuitable for such service overlay networks.

C. Generalized Traffic Split Ratios

The traffic split ratios α_i can be generalized to depend on *source or destination nodes* of the traffic, or both. We discuss the latter version here. While this generalization does not meet the indirection requirement of service overlays like i3, it can potentially increase the throughput performance of the two-phase routing scheme for other application scenarios like IP-over-Optical networks.

Suppose that a fraction α_k^{ij} of the traffic that originates at node i whose destination is node j is routed to node k in Phase 1. The traffic split ratios associated with any source-destination pair must sum to unity, i.e., $\sum_{k \in N} \alpha_k^{ij} = 1$ for all $i, j \in N$. Let us compute the total demand that is needed between nodes a and b to route Phase 1 and Phase 2 paths. Let the current traffic matrix be $T = [t_{ij}] \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$. In the first phase, a fraction α_b^{ak} of the traffic t_{ak} originating at node a and destined for node k is sent to intermediate node b. Thus, the demand from node a to node b for Phase 1 traffic

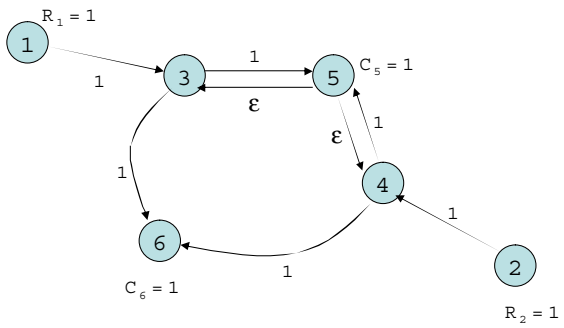

Fig. 3. 6-node network to illustrate throughput improvement with generalized traffic split ratios for Two-Phase Routing.

is $\sum_{k\in N} \alpha_b^{ak} t_{ak}$. A fraction α_a^{kb} of the traffic t_{kb} originating at node k and destined for node b is sent to intermediate node a in Phase 1 and needs to be routed to node b in the second phase. Thus, the demand from node a to node b for Phase 2 traffic is $\sum_{k\in N} \alpha_a^{kb} t_{kb}$. Therefore, the total demand τ_{ab} that needs to be statically provisioned from node a to node b is the maximum value, taken over all traffic matrices $T \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$, of the sum of the above two quantities, that is,

$$\tau_{ab} = \max_{[t_{ij}]\in\mathcal{T}(\vec{\mathcal{R}},\vec{\mathcal{C}})} \left[\sum_{k\in N} \alpha_b^{ak} t_{ak} + \sum_{k\in N} \alpha_a^{kb} t_{kb} \right] \qquad (14)$$

The quantity above appears to involve bilinear terms but can be very nicely accommodated into an LP. In fact, as one of the key contributions of this paper, we give the first polynomial size LP for maximum throughput two-phase routing of hose traffic with generalized traffic split ratios.

Before proceeding further, we give an example to illustrate the improvement in throughput when we generalize the traffic split ratios as above. Consider the 6-node network shown in Figure 3. Here $R_1 = R_2 = 1$ and $C_5 = C_6 = 1$. All other R_i, C_j values are zero. The capacities of links $(5, 3)$ and $(5, 4)$ are each equal to some small quantity $\epsilon > 0$. All other links shown have unit capacity.

Observe that node 1 has a unit capacity path to node 3 but the capacity of the path to node 4 is small (= ϵ). Similarly, node 2 has a unit capacity path to node 4 but the capacity of the path to node 3 is small (= ϵ). Thus, when maximizing throughput, node 4 is not a good choice for serving as intermediate node for the traffic originating at node 1. Similarly, node 3 is not a good choice for serving as intermediate node for the traffic originating at node 2. If the traffic split ratios are dependent on intermediate nodes *only* (and not on source or destination of traffic), the throughput of two-phase routing will be small. By making the traffic split ratios dependent on the source of traffic also, two-phase routing can completely avoid routing along the links with small capacities. In fact, the gap between the throughputs of two-phase routing with intermediate node dependent traffic split ratios α_k and generalized traffic split ratios α_k^{ij} can be made arbitrarily large by making the value of ϵ arbitrarily small.

However, in view of the 2-optimality result for two-phase routing in Section III-E that uses only intermediate node dependent traffic split ratios and assumes $R_i = C_i$ for all i, it follows that such pathological examples where the throughput improvement with generalized split ratios is arbitrarily large (or, even greater than 2) do not exist when ingress-egress capacities are symmetric.

D. Throughput Maximization

Given a network with link capacities u_e and constraints R_i, C_i on the ingress-egress traffic, we consider the problem of two-phase routing with generalized traffic split ratios so as to maximize the network throughput. The throughput is the maximum multiplier λ such that all matrices in $\lambda \cdot \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$ can be feasibly routed under given link capacities.

We begin with an LP formulation with an infinite number of constraints and a polynomial size separation oracle LP for it, and then combine the two into a polynomial size LP that can be solved in polynomial time using a general linear programming algorithm [14].

LP with Infinite Constraints and Separation Oracle

The routing of d_{ab} amount of traffic, as given by equation (14) above, for each source-destination pair (a, b) can be specified by a set of *flow variables* x_e^{ab}, where x_e^{ab} denotes the amount of traffic from node a to node b that traverses link e in the network. Let μ denote the maximum utilization of any link in the network. Maximizing the throughput is equivalent to minimizing the maximum link utilization μ.

$$\text{minimize} \quad \mu$$

subject to

$$\tau_{ab} \geq \sum_{k\in N} \alpha_b^{ak} t_{ak} + \sum_{k\in N} \alpha_a^{kb} t_{kb}$$
$$\forall\ [t_{ij}] \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}}),\ \forall\ a, b \in N \qquad (15)$$

$$\sum_{e\in E^+(k)} x_e^{ij} - \sum_{e\in E^-(k)} x_e^{ij} = \begin{cases} +\tau_{ij} & \text{if } k = i \\ -\tau_{ij} & \text{if } k = j \\ 0 & \text{otherwise} \end{cases}$$
$$\forall\ i, j, k \in N \qquad (16)$$

$$\sum_{i,j\in N} x_e^{ij} \leq \mu u_e\ \forall\ e \in E \qquad (17)$$

$$\sum_{k\in N} \alpha_k^{ab} = 1\ \forall\ a, b \in N \qquad (18)$$

$$\alpha_k^{ab} \geq 0\ \forall\ k, a, b \in N \qquad (19)$$

$$x_e^{ij} \geq 0\ \forall\ e \in E,\ \forall\ i, j \in N \qquad (20)$$

Constraints (15) correspond to the value of the demand τ_{ab} from node a to node b as given in equation (14). Constraint (16) correspond to routing of flows between each source-destination pair of the required value. Constraints (17) are the maximum utilization constraints for each link. Constraints (18) correspond to the traffic split ratios summing to 1 for each source-destination pair. The quantities t_{ij} in the RHS of (15) are constants and hence the constraints are linear. Note that there are an infinite set of constraints in (15), since there are $n(n-1)$ constraints for each $[t_{ij}] \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$.

The above LP can be solved in polynomial time by the ellipsoid algorithm [14] provided we can find a polynomial time separation oracle for the constraints (15). Given a set of values for the variables in the above LP, the separation oracle needs to identify at least one constraint that is violated (if any), or indicate otherwise. Clearly, constraints (16)-(18) can be verified in polynomial time.

To determine if the constraints in (15) are violated for any link, we need to either identify a source-destination pair (a, b) and a traffic matrix $[t_{ij}] \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$ such that the corresponding constraint is violated, or determine that all such constraints are satisfied. This can be done by verifying that for each source-destination pair (a, b), the LP below, with variables $t_{ij}\ \forall\ i, j \in N$, has optimum objective function value at most τ_{ab}. If not, the traffic matrix $[t_{ij}]$ obtained in the optimal solution of the LP identifies the corresponding violating constraint in (15).

$$\text{maximize} \quad \sum_{k \in N} \alpha_b^{ak} t_{ak} + \sum_{k \in N} \alpha_a^{kb} t_{kb}$$

subject to

$$\sum_{j \in N, j \neq i} t_{ij} \leq R_i\ \forall\ i \in N \quad (21)$$

$$\sum_{i \in N, i \neq j} t_{ij} \leq C_j\ \forall\ j \in N \quad (22)$$

$$t_{ij} \geq 0\ \forall\ i, j \in N \quad (23)$$

As noted earlier, the ellipsoid algorithm gives running times that are not feasible for practical implementations. Hence, the motivation for designing a polynomial size LP for the above problem. Such an LP can be directly fed into LP solvers like CPLEX [4] for solution.

Polynomial Size LP

In developing the polynomial size LP, we first take the dual of the separation oracle LP above. For a given source-destination pair (a, b), the dual LP has non-negative variables $r(i, a, b)$ corresponding to each constraint in (21) and non-negative variables $c(j, a, b)$ corresponding to each constraint in (22).

$$\text{minimize} \quad \sum_{i \in N} R_i r(i, a, b) + \sum_{j \in N} C_j c(j, a, b)$$

subject to

$$r(a, a, b) + c(b, a, b) \geq \alpha_a^{ab} + \alpha_b^{ab} \quad (24)$$

$$r(a, a, b) + c(k, a, b) \geq \alpha_b^{ak}\ \forall\ k \in N, k \neq b \quad (25)$$

$$r(k, a, b) + c(b, a, b) \geq \alpha_a^{kb}\ \forall\ k \in N, k \neq a \quad (26)$$

$$r(i, a, b), c(i, a, b) \geq 0\ \forall\ i \in N \quad (27)$$

It follows directly from strong duality of linear programming [14] that for each source-destination pair (a, b), the primal (separation oracle) LP has an optimum objective function value of at most τ_{ab} if and only if the dual LP has a feasible solution with objective function value at most τ_{ab}.

The requirement that the dual LPs, for all $a, b \in N$, have feasible solutions with objective function value at most τ_{ab} can be modeled as the following constraint:

$$\sum_{i \in N} R_i r(i, a, b) + \sum_{j \in N} C_j c(j, a, b) \leq \tau_{ab}\ \forall\ a, b \in N$$

This allows us to remove the infinite set of constraints in (15) and add the above constraint and constraints (24)-(27) from the dual LPs to obtain the following polynomial size LP for our problem:

$$\text{minimize} \quad \mu$$

subject to

$$\sum_{e \in E^+(k)} x_e^{ij} - \sum_{e \in E^-(k)} x_e^{ij} = \begin{cases} +\tau_{ij} & \text{if } k = i \\ -\tau_{ij} & \text{if } k = j \\ 0 & \text{otherwise} \end{cases} \quad \forall\ i, j, k \in N \quad (28)$$

$$\sum_{i,j \in N} x_e^{ij} \leq \mu u_e\ \forall\ e \in E \quad (29)$$

$$\sum_{k \in N} \alpha_k^{ab} = 1\ \forall\ a, b \in N \quad (30)$$

$$\sum_{i \in N} R_i r(i, a, b) + \sum_{j \in N} C_j c(j, a, b) \leq \tau_{ab}\ \forall\ a, b \in N \quad (31)$$

$$r(a, a, b) + c(b, a, b) \geq \alpha_a^{ab} + \alpha_b^{ab} \quad \forall\ a, b \in N \quad (32)$$

$$r(a, a, b) + c(k, a, b) \geq \alpha_b^{ak}\ \forall k, a, b \in N \quad (33)$$

$$r(k, a, b) + c(b, a, b) \geq \alpha_a^{kb}\ \forall k, a, b \in N \quad (34)$$

$$\alpha_k^{ab} \geq 0\ \forall\ k, a, b \in N \quad (35)$$

$$x_e^{ij} \geq 0\ \forall\ e \in E, \quad \forall\ i, j \in N \quad (36)$$

$$r(i, a, b), c(i, a, b) \geq 0\ \forall\ i, a, b \in N \quad (37)$$

This LP has $n^2(n-1)$ constraints in (28), m constraints in (29), $n(n-1)$ constraints each in (30)-(32), $n(n-1)(n-2)$ constraints each in (33)-(34), $n^2(n-1)$ constraints in (35), $mn(n-1)$ constraints in (36), and $2n^2(n-1)$ constraints in (37), for a total of $O(mn^2)$ constraints. The number of variables is $n^2(n-1)+n(n-1)+mn(n-1)+2n^2(n-1)+1 = O(mn^2)$.

By using per-source flow variables x_e^i instead of per source-destination variables x_e^{ij}, the number of variables and constraints in the above LP can be reduced to $O(n^3)$.

E. Optimality Bound for Two-Phase Routing

Two-phase routing specifies ratios for splitting traffic among intermediate nodes and Phase 1 and Phase 2 paths for routing them. Thus, two-phase routing is one form of fixed path routing. However, as explained in Section III-B, it has the desirable property of static provisioning that a general solution of fixed path routing (e.g., direct source-destination path routing) may not have. Moreover, when the traffic split ratios in two-phase routing depend on intermediate nodes only, the scheme does

not require a packet's final destination to be known as the source, an indirection property that is required of specialized service overlays like i3.

The subject of investigation in this paper is: *Do the desirable properties of two-phase routing come with any resource (throughput) overhead compared to (i) direct source-destination path routing, and (ii) optimal scheme among the class of all schemes that are allowed to make the routing dynamically dependent on the traffic matrix?* We address this question from two approaches.

First, using the polynomial size LP formulations developed in Sections II-A and III-D, we compare the throughput of two-phase routing with that of direct source-destination path routing on actual ISP topologies in Section V. Using upper bounds on the throughput of the optimal scheme computed as discussed in Section IV, we compare the throughput of two-phase routing with that of the optimal scheme.

Second, we analyze the throughput requirements of two-phase routing from a theoretical perspective and establish a 2-optimal bound in this section. That is, the throughput of two-phase routing is at least $1/2$ that of the best possible scheme in which the *routing can be dependent on the traffic matrix*. We would like to emphasize the generality of this result – it compares two-phase routing with the *most general class of schemes* for routing hose traffic.

Characterization of Optimal Scheme

Consider the class of all schemes for routing all matrices in $\mathcal{T}(\vec{\mathcal{R}},\vec{\mathcal{C}})$ where the routing can be made dependent on the traffic matrix. For any scheme $\mathcal{A}$, let $A(e,T)$ be the traffic on link e when matrix T is routed by $\mathcal{A}$. Then, the throughput λ_A of scheme $\mathcal{A}$ is given by

$$\lambda_A = \min_{e\in E} \frac{u_e}{\max_{T\in\mathcal{T}(\vec{\mathcal{R}},\vec{\mathcal{C}})} A(e,T)}$$

The optimal scheme is the one that achieves the maximum throughput λ_{OPT} among all schemes. This is given by

$$\lambda_{OPT} = \max_{\mathcal{A}} \lambda_A$$

In the following lemma, the throughput of the optimal scheme is expressed in another way. For each $T \in \mathcal{T}(\vec{\mathcal{R}},\vec{\mathcal{C}})$, let $\lambda(T)$ be the maximum throughput achievable for routing the single matrix T.

Lemma 1: The throughput of the optimal scheme is given by

$$\lambda_{OPT} = \min_{T\in\mathcal{T}(\vec{\mathcal{R}},\vec{\mathcal{C}})} \lambda(T)$$

Proof: For any matrix $T \in \mathcal{T}(\vec{\mathcal{R}},\vec{\mathcal{C}})$, since the optimal scheme has to route it, the quantity λ_{OPT} is upper bounded by $\lambda(T)$. Thus, $\lambda_{OPT} \leq \min_{T\in\mathcal{T}(\vec{\mathcal{R}},\vec{\mathcal{C}})} \lambda(T)$. This minimum throughput can indeed be achieved by routing every matrix in a way that maximizes its throughput. Hence equality holds in the above upper bound. ∎

At first glance, the optimal scheme that maximizes throughput appears to be hard to specify because it can route each traffic matrix differently, of which there are infinitely many in $\mathcal{T}(\vec{\mathcal{R}},\vec{\mathcal{C}})$. However, because the link capacities are given in our throughput maximization model, (an) the optimal scheme can be characterized in a simple way from the proof of the lemma. Given a traffic matrix as input, route it in a manner that maximizes its throughput. Routing a single matrix so as to maximize its throughput is also known as the *maximum concurrent flow problem* [15] and is solvable in polynomial time. Clearly, the routing is dependent on the traffic matrix and can be different for different matrices.

The problem of computing λ_{OPT} can be shown to be $co\mathcal{NP}$-hard. Computing the cost of the optimal scheme for the minimum cost network design version of the problem is also known to be $co\mathcal{NP}$-hard – the result is stated without proof in [8]. (An) The optimal scheme for minimum cost network design does not even appear to have a simple characterization like that for maximum throughput network routing.

2-Optimality Result for Two-Phase Routing

The 2-optimal bound for two-phase routing that we prove next establishes that two-phase routing provides a 2-approximation to the optimal scheme *for both maximum throughput network routing and minimum cost network design*. We view this as an important theoretical contribution, given the computational intractability of the optimal schemes for both problems.

Even though this theoretical result shows that the throughput of two-phase routing, in the *worst case*, can be as low as $1/2$ that of the optimal scheme (and, hence that of direct source-destination path routing), the experiments in Section V indicate that two-phase routing performs much better in practice – *the throughput of two-phase routing matches that of direct source-destination path routing and is within 6% of that of the optimal scheme on all evaluated topologies.*

We assume that $R_i = C_i$ for all nodes i. This is not a restrictive assumption because network routers and switches have bidirectional ports (line cards), hence the ingress and egress capacities are equal.

Theorem 1: Let $R_i = C_i$ for all nodes i, and $R = \sum_{i\in N} R_i$. Consider the throughput maximization problem under given link capacities. Then, the throughput of the optimal scheme is at most

$$2\left(1 - \frac{1}{R}\min_{i\in N} R_i\right)$$

times that of two-phase routing.

Proof: Let α_i be the traffic split ratios associated with each intermediate node i in two-phase routing. Set $\alpha_i = \frac{R_i}{R}$ for all $i \in N$. Then, the demand matrix $D = [d_{ij}]$ as a result of two phase routing is given by

$$\begin{aligned} d_{ij} &= \alpha_j R_i + \alpha_i C_j \\ &= \alpha_j R_i + \alpha_i R_j \\ &= 2\frac{R_i R_j}{R} \end{aligned}$$

for all $i \neq j$ and $d_{ii} = 0$ for all i.

Now consider the traffic matrix $T = [t_{ij}]$ where

$$t_{ij} = \frac{R_i R_j}{R}$$

for all $i \neq j$ and $t_{ii} = 0$ for all i. Let β be the maximum multiplier such that $\beta T \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$. Then, we must have

$$\begin{aligned} \beta \sum_{j \in N, j \neq i} t_{ij} &\leq R_i \ \forall \ i \in N \\ \beta \sum_{j \in N, j \neq i} \frac{R_i R_j}{R} &\leq R_i \ \forall \ i \in N \\ \beta \frac{R_i(R - R_i)}{R} &\leq R_i \ \forall \ i \in N \\ \beta &\leq \frac{R}{R - R_i} \ \forall \ i \in N \end{aligned}$$

whence,

$$\beta = \frac{R}{R - \min_{i \in N} R_i}$$

Since $D = 2T$, hence $\beta T = \frac{\beta}{2} D$. Since the optimal scheme must route the matrix $\beta T \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$, its throughput is at most the throughput $\lambda(\beta T)$ for routing matrix βT (using Lemma 1). Hence,

$$\lambda_{OPT} \leq \lambda(\beta T) = \lambda(\frac{\beta}{2} D) = \frac{2}{\beta} \lambda(D)$$

The last step uses the property that the throughput of c times a given matrix is equal to $\frac{1}{c}$ times the throughput of the original matrix. Since D is the demand matrix for two-phase routing, we conclude that the throughput of the optimal scheme is at most

$$\frac{2}{\beta} = 2 \left(1 - \frac{1}{R} \min_{i \in N} R_i \right)$$

times that of two-phase routing. ■

The following result for minimum cost network design follows from an argument similar to that for Theorem 1.

Theorem 2: Let $R_i = C_i$ for all nodes i, and $R = \sum_{i \in N} R_i$. Consider the minimum cost network design problem under given link costs for unit traffic. Then, the cost of two-phase routing is at most

$$2 \left(1 - \frac{1}{R} \min_{i \in N} R_i \right)$$

times that of the optimal scheme.

IV. Upper Bounding Throughput of Optimal Scheme

In this section, we discuss a method for upper bounding the throughput of the optimal scheme among the class of schemes that are allowed to reconfigure the routing with changes in the traffic matrix. In view of the remarks that we made about the computational intractability of the throughput of the optimal scheme, this upper bound will be useful in comparing the throughput of two-phase routing and direct source-destination path routing with that of the optimal scheme.

From Lemma 1, we have $\lambda_{OPT} = \min_{T \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})} \lambda(T)$. Thus, we would a like to identify a matrix $T \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$ for which $\lambda(T)$ is minimum. This matrix T is hard to compute. Suppose that we take any single matrix $T \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$ and compute its maximum throughput $\lambda(T)$ – the maximum throughput for routing a single matrix under given link capacities can be solved using the maximum concurrent flow problem [15]. This certainly gives an upper bound on λ_{OPT}, since $\lambda_{OPT} \leq \lambda(T)$. We use a heuristic approach to find a matrix that gives tight upper bounds.

Consider a matrix T with throughput $\lambda(T)$ whose maximum throughput routing uses x_e capacity on link e. Since $\lambda(T) x_e \leq u_e$ for all e, we have $\sum_{e \in E} \lambda(T) x_e \leq \sum_{e \in E} u_e$, whence

$$\lambda_{OPT} \leq \lambda(T) \leq \frac{\sum_{e \in E} u_e}{\sum_{e \in E} x_e}$$

Let $B(T)$ be the minimum bandwidth required to route matrix T. Then, $\sum_{e \in E} x_e \geq B(T)$, whence $\lambda_{OPT} \leq \frac{\sum_{e \in E} u_e}{B(T)}$. Thus, the least upper bound obtained in this manner is given by

$$\lambda_{OPT} \leq \frac{\sum_{e \in E} u_e}{\max_{T \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})} B(T)} \tag{38}$$

The matrix $T \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$ that takes the highest bandwidth to route can be computed in polynomial time as follows. The minimum bandwidth routing must route all demands *along shortest hop paths*. Let d_{ij} denote the hop count of a shortest path from node i to j for all $i, j \in N$. Then, the problem of determining the traffic matrix $T = [t_{ij}] \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$ that takes the maximum bandwidth to route can be formulated as the following linear program:

$$\text{maximize} \quad \sum_{i,j \in N} d_{ij} t_{ij}$$

subject to

$$\begin{aligned} \sum_{j \in N, j \neq i} t_{ij} &\leq R_i \ \forall \ i \in N & (39) \\ \sum_{i \in N, i \neq j} t_{ij} &\leq C_j \ \forall \ j \in N & (40) \\ t_{ij} &\geq 0 \ \forall \ i, j \in N & (41) \end{aligned}$$

The required bandwidth $B(T)$ is the objective function of the linear program and the ingress-egress traffic capacities that define $\mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})$ form the constraints.

Let the optimum solution to this linear program be the matrix T^*. The value of $B(T^*) = \max_{T \in \mathcal{T}(\vec{\mathcal{R}}, \vec{\mathcal{C}})} B(T)$ thus obtained gives us an upper bound on λ_{OPT} using inequality (38) above. Note that maximum throughput routing does not necessarily route along shortest paths. Hence, we can actually compute the throughput $\lambda(T^*)$ of the matrix T^* and check if that gives a better (lower) upper bound (since $\lambda_{OPT} \leq \lambda(T^*)$). In all experiments, the latter gave a better upper bound.

V. Throughput Comparisons

In this section, we compare the throughput performance of four schemes for routing hose traffic, namely, (i) two-phase routing with intermediate node dependent traffic split ratios α_k, (ii) two-phase routing with generalized traffic split ratios α_k^{ij}, (iii) direct source-destination path routing, and (iv) optimal scheme. For (i), we use the linear programming formulation with α_i traffic split ratios. For (ii) and (iii), we use the linear programming formulations developed in this paper. We use the method discussed in Section IV to obtain an upper bound for λ_{OPT}. We use CPLEX [4] to solve all linear programs.

Topology	Routers (original)	Links (inter-router)	PoPs (coalesced)	Links (inter-PoP)
Telstra (Australia) 1221	108	306	57	59
Sprintlink (US) 1239	315	1944	44	83
Ebone (Europe) 1755	87	322	23	38
Tiscali (Europe) 3257	161	656	50	88
Exodus (Europe) 3967	79	294	22	37
Abovenet (US) 6461	141	748	22	42

TABLE I

ROCKETFUEL TOPOLOGIES: ORIGINAL NUMBER OF ROUTERS AND INTER-ROUTER LINKS, AND NUMBER OF COALESCED POPS AND INTER-POP LINKS.

A. Topologies and Link/Ingress-Egress Capacities

For our experiments, we use six ISP topologies collected by Rocketfuel, an ISP topology mapping engine [16]. These topologies list multiple intra-PoP (Point of Presence) routers and/or multiple intra-city PoPs as individual nodes. We coalesced PoPs into nodes corresponding to cities so that the topologies represent geographical PoP-to-PoP ISP topologies. Some data about the original Rocketfuel topologies and their coalesced versions is provided in Table I.

Link capacities, which are required to compute the maximum throughput, are not available for these topologies. Rocketfuel computed OSPF/IS-IS link weights for the topologies so that shortest cost paths match observed routes. In order to deduce the link capacities from the weights, we assumed that the given link weights are the default setting for OSPF weights in Cisco routers, i.e., inversely proportional to the link capacities [3]. The link capacities obtained in this manner turned out to be symmetric, i.e., $u_{ij} = u_{ji}$ for all $(i, j) \in E$.

There is also no available information on the ingress-egress traffic capacities at each node. Because ISPs commonly engineer their PoPs to keep the ratio of add/drop and transit traffic approximately fixed, we assumed that the ingress-egress capacity at a node is proportional to the total capacity of network links incident at that node. We also assume that $R_i = C_i$ for all nodes i – since network routers and switches have bidirectional ports (line cards), hence the ingress and egress capacities are equal. Thus, we have $R_i (= C_i) \propto \sum_{e \in E^+(i)} u_e$.

B. Experiments and Results

We denote the throughput values for the three different schemes as follows: (i) λ_{TPR} for two-phase routing with intermediate node dependent traffic split ratios, (ii) λ_{GTPR} for two-phase routing with generalized traffic split ratios, and (iii) λ_{DPR} for direct source-destination path routing. Clearly, $\lambda_{TPR} \leq \lambda_{GTPR} \leq \lambda_{DPR} \leq \lambda_{OPT}$.

The quantity $\lambda_{TPR}/\lambda_{OPT}$ gives the closeness of the throughput performance of two-phase routing (with α_i traffic split ratios) to that of the optimal scheme. A lower bound on this quantity, expressed as a percentage, is listed in Table I for the six Rocketfuel topologies. The lower bound on $\lambda_{TPR}/\lambda_{OPT}$ is obtained by using the method discussed in Section IV to compute an upper bound for λ_{OPT}. The table shows that the throughput performance of two-phase routing with intermediate node dependent split ratios α_i is within 6% of that of the optimal scheme. (For two of the topologies, it actually matches that of the optimal scheme.) Hence, for these topologies, there is clearly not much room for throughput improvement when we generalize the traffic split ratios in two-phase routing or even consider arbitrary fixed path routing solutions.

Topology	Lower bound on $\lambda_{TPR}/\lambda_{OPT}$
Telstra (Australia) 1221	100%
Sprintlink (US) 1239	97.71%
Ebone (Europe) 1755	98.90%
Tiscali (Europe) 3257	95.65%
Exodus (Europe) 3967	100%
Abovenet (US) 6461	94.82%

TABLE II

CLOSENESS OF THROUGHPUT OF TWO-PHASE ROUTING (λ_{TPR}) TO THAT OF THE OPTIMAL SCHEME (λ_{OPT}) FOR ROCKETFUEL TOPOLOGIES.

For the Tiscali 3257 topology, the CPLEX processes for solving the linear programs for λ_{GTPR} and λ_{FPR} ran out of memory and were killed on a 2.4GHz Dual Xeon machine with 1GB of RAM and running Linux. This was the fastest machine with the highest RAM that we had access to for running CPLEX. For the Exodus 3967 and Telstra 1221 topologies, the throughput of two-phase routing with traffic split ratios α_i matches that of the optimal scheme (as reported in Table I), hence $\lambda_{TPR} = \lambda_{GTPR} = \lambda_{DPR}$. The latter was observed to be the case with the remaining three Rocketfuel topologies also.

Thus, on five of the six Rocketfuel topologies, the *throughput of two-phase routing with α_i traffic split ratios equals that with generalized traffic split ratios and matches the throughput of direct source-destination routing along fixed paths*. (Recall that the pathological example for the improvement in throughput of two-phase routing with generalized traffic split ratios in Section III-C exploited $R_i \neq C_i$ for some node nodes i and asymmetric link capacities – both of these are not present in the Rocketfuel topologies.)

Given the identical throughput performance of the two versions of two-phase routing, the simpler version with intermediate node-dependent traffic split ratios α_i is preferred because of its ability to support indirection in specialized service overlay models like i3.

These experiments on actual ISP topologies indicate that two-phase routing achieves its robustness to traffic variation without compromising throughput performance compared to previous approaches like direct source-destination path routing. Its throughput performance is within 6% of that of

the optimal scheme on the evaluated topologies. Thus, two-phase routing is able to handle highly variable traffic in a capacity efficient manner and provide the desirable properties of (i) static provisioning at the optical layer in IP-over-Optical networks, and (ii) supporting indirection in specialized service overlay networks. Direct source-destination routing does not meet these requirements.

VI. Conclusion and Future Work

The two-phase routing scheme was recently proposed for routing highly dynamic and changing traffic patterns on the Internet with QoS guarantees. If deployed, it will allow service providers to operate their networks in a quasi-static manner where both intra-domain paths and the bandwidths allocated to these paths is robust to extreme traffic variation. The scheme has the desirable properties of supporting (i) static optical layer provisioning in IP-over-Optical networks, and (ii) indirection in specialized service overlay models like i3. These are not supported by other approaches for routing hose traffic, such as direct source-destination routing along fixed paths.

In this paper, we evaluated the throughput performance of two-phase routing. Quite surprisingly, experiments on actual ISP topologies taken from the RocketFuel project show that the throughput of generalized two-phase routing matches that of direct source-destination path routing. Also, two-phase routing with generalized traffic split-ratios has the same throughput as that for intermediate node-dependent traffic split ratios. Given the identical throughput performance of the two versions of two-phase routing, the simpler version with intermediate node-dependent traffic split ratios is preferred because of its ability to support indirection in service overlay models like i3. Also, the throughput of two-phase routing is within 6% of that of the optimal scheme on all evaluated topologies.

This conclusion should lead to increased acceptance of a two-phase routing based architecture for routing highly variable traffic and mitigate any concerns that the desirable properties of two-phase routing come with substantial over-provisioning of capacity. We believe that establishment of the capacity efficiency of two-phase routing through rigorous investigation, as attempted in this paper, will influence and shape the mindset of ISPs in considering its deployment in actual networks.

In the course of our investigation, we also made several theoretical contributions. First, we developed the first polynomial size LP formulation for maximum throughput routing of hose traffic along direct source-destination paths. Second, we developed the first polynomial size LP formulation for maximum throughput two-phase routing of hose traffic with generalized traffic split ratios. Third, we proved that the throughput of two-phase routing is at least $1/2$ that of the optimal scheme among the class of all schemes that are allowed to make the routing dynamically dependent on the traffic matrix.

There exist many opportunities to extend our current work. On the theoretical front, it might be possible to improve the 2-optimality throughput bound of two-phase routing under special assumptions, e.g., small-degree graphs. The motivation for this is the empirical evidence that the performance of two-phase routing is much better on ISP topologies – these are characterized by small nodal degree, as was true for all the Rocketfuel topologies considered in this paper.

The observation that the two versions of two-phase routing – intermediate node dependent and generalized traffic split ratios – and direct source-destination path routing have identical throughput performance on the evaluated topologies merits further theoretical investigation. In particular, it would be interesting to identify the assumptions under which this might be universally true (e.g., symmetric ingress-egress and link capacities).

References

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*, Prentice Hall, February 1993.

[2] Y. Azar, E. Cohen, A. Fiat, H. Kaplan, and H. Räcke, "Optimal oblivious routing in polynomial time", *35th ACM Symposium on the Theory of Computing (STOC)*, 2003.

[3] "Configuring OSPF", Cisco Systems Product Documentation, http://www.cisco.com/univercd/home/home.htm.

[4] ILOG CPLEX, http://www.ilog.com.

[5] N. G. Duffield, P. Goyal, A. G. Greenberg, P. P. Mishra, K. K. Ramakrishnan, J. E. van der Merwe, "A flexible model for resource management in virtual private network", *ACM SIGCOMM 1999*, August 1999.

[6] T. Erlebach and M. Rüegg, "Optimal Bandwidth Reservation in Hose-Model VPNs with Multi-Path Routing", *IEEE Infocom 2004*, March 2004.

[7] J. A. Fingerhut, S. Suri, and J. S. Turner, "Designing Least-Cost Nonblocking Broadband Networks", *Journal of Algorithms*, 24(2), pp. 287-309, 1997.

[8] A. Gupta, J. Kleinberg, A. Kumar, R. Rastogi, B. Yener, "Provisioning a Virtual Private Network: A Network Design Problem for Multicommodity Flow", *ACM Symposium on Theory of Computing (STOC) 2001*, July 2001.

[9] M. Kodialam, T. V. Lakshman, and S. Sengupta, "Efficient and Robust Routing of Highly Variable Traffic", *Third Workshop on Hot Topics in Networks (HotNets-III)*, November 2004.

[10] A. Kumar, R. Rastogi, A. Silberschatz , B. Yener, "Algorithms for provisioning VPNs in the hose model", *ACM SIGCOMM 2001*, August 2001.

[11] O. L. Mangasarian, "Linear and Nonlinear Separation of Patterns by Linear Programming", *Operations Research 13*, 1965, pp. 444-452.

[12] A. Medina, N. Taft, K. Salamatian, S. Bhattacharyya, C. Diot, "Traffic Matrix Estimation: Existing Techniques and New Directions", *ACM SIGCOMM 2002*, August 2002.

[13] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, January 2001.

[14] A. Schrijver, *Theory of Linear and Integer Programming*, John Wiley & Sons, 1986.

[15] F. Shahrokhi and D. Matula, "The Maximum Concurrent Flow Problem", *Journal of ACM*, 37(2):318-334, 1990.

[16] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson, "Measuring ISP Topologies with Rocketfuel", *IEEE/ACM Transactions on Networking*, vol. 12, no. 1, pp. 2-16, February 2004.

[17] I. Stoica, D. Adkins, S. Zhuang, S. Shenker, S. Surana, "Internet Indirection Infrastructure", *ACM SIGCOMM 2002*, August 2002.

[18] R. Zhang-Shen and N. McKeown "Designing a Predictable Internet Backbone Network", *Third Workshop on Hot Topics in Networks (HotNets-III)*, November 2004.

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the Proceedings IEEE Infocom.