

Effective Usage of Dynamic Circuits for IP Routing

Mohit Chamania, Marcel Caria and Admela Jukan
Technische Universität Carolo-Wilhelmina zu Braunschweig
Email: chamania, caria, jukan@ida.ing.tu-bs.de

Abstract—The Internet is susceptible to congestion due to progressively increasing traffic as well as short-lived traffic surges. Traditional mechanisms to counter these effects over-provision the IP network to a significant degree. At the same time, the advent of dynamic circuits in L2/L1 networks has fueled the research in the area of network engineering which deals with the ability to add capacity in the higher layer (IP) by establishing dynamic circuits in the lower layers. However, effective usage of dynamic circuits is a challenge, as they can lead to IP routing instabilities. We present an approach wherein IP bypass links are established using dynamic circuits to alleviate congestion while keeping the routing stable in the IP layer. Proposed is an ILP based approach which computes the optimal set of circuits with and without the knowledge of the traffic matrix in the IP layer. The results show that even without the knowledge of the traffic matrix, the proposed method computes only a marginally higher number of bypasses, albeit at a higher capacity.

I. INTRODUCTION

The Internet is known to be susceptible to sudden surges in traffic which can be triggered by various natural, malicious, or commercial events and can significantly hamper the performance of IP networks. To counter this, a *headroom* practice is employed typically, where the IP network is significantly over-provisioned. As an example, the Internet2 network maintains around 75 percent headroom in all IP links [1], implying that the link utilization maybe less than 25 percent. While the practice of large headroom allows for accommodation of unpredictable large-bandwidth applications and also keeps the network stable and resilient against attacks and link failures, the significant over-provisioning also leads to high CAPEX and OPEX of higher speed IP interfaces as well as higher energy consumption in the network.

Recent efforts to counter the over-provisioning of IP links have suggested to establish dynamic *circuits* between IP routers in order to either increase capacity of already existing IP links, or to create new links in the IP layer which are then used to re-route traffic. However, despite the myriad of proposed methods, such as “IP-over-optical” solutions, Internet Service Providers (ISPs) remain reluctant to deploy dynamic links for frequent operations, as the frequent introduction and deletion of IP links can lead to routing changes and severe instabilities in network management. While routing changes alone may not cause fatal service disruptions, additional network management tasks associated with the same, such as flow monitoring, event correlation for alarm suppression and fault management, must be reconfigured in the network upon every link setup. Such network management reconfigurations are critical, tedious and error-prone, and therefore fundamentally unsuitable for commercial ISPs.

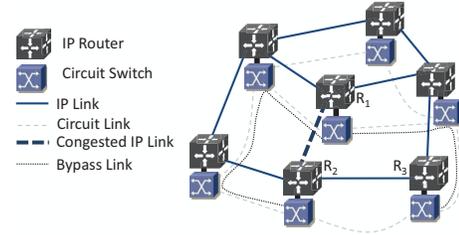


Fig. 1. A multi-layer topology with congested links demonstrating the use of the proposed bypass mechanism

In this work, we propose a new dynamic circuit paradigm, we refer to as *optical bypass*, with the goal to keep IP routing stable and eliminate problems such as route flapping, while alleviating traffic congestion and over-provisioning. In our approach, we add *bypasses* without advertising them in the IP network, and the bypasses are only visible to the corresponding ingress and egress routers. We then select flows to be rerouted over these bypasses with a constraint that both the ingress and egress of the bypass be on the original IP routing path for that flow; note here that a flow refers to aggregate application flows between two routers. We present two models for the bypass computation: with and without knowledge of the traffic matrix. The latter is of significant practical interest, as determining the IP traffic matrix is a non-trivial problem. To determine the bypasses with incomplete traffic information, we propose to use alternate traffic measurements, which are commercially available and do not require proprietary implementations.

Consider the example network presented in Fig. 1: the link $R_1 - R_2$ is congested, and a bypass is established from R_1 to R_3 to divert traffic from this link. By our constraints, the bypass from R_1 to R_3 can only be used to reroute flows which have R_1 and R_3 in their original routing path (here the flow from $R_1 - R_3$) and therefore cannot be used to re-route traffic from R_1 to R_2 . The constraint is used to ensure that no existing links in the network observe an increase in traffic. For instance, if the traffic from R_1 to R_2 was bypassed from R_1 to R_3 , the link from R_3 to R_2 would experience an increase in traffic, and is avoided. Our proposal is targeted to handle frequent short term traffic churns and therefore gives more weightage to routing stability as compared to congestion due to long term effects which focus on resource optimization.

The rest of the paper is organized as follows: In section II, we present related work and motivation behind the study. Section III and IV present the ILP based approach for networks with known and unknown traffic matrix, respectively, to compute the optimal bypass topology. Section V presents numerical results comparing the proposed approaches, while section VI concludes the paper.

II. RELATED WORK AND OUR CONTRIBUTION

Efforts to counter over-provisioning of IP links can be categorized into two complimentary approaches: Traffic Engineering (TE) and Network Engineering (NE) [2]. Whereas traffic engineering attempts to optimize the routing of IP traffic, network engineering adds additional capacity to the IP network when TE mechanisms cannot cope with increasing traffic. Most NE proposals assume knowledge of the complete IP traffic matrix and attempt to *minimize the cost* of changing the IP network topology or the *number of topology reconfigurations* using ILP/heuristic based solutions [3], [4] or genetic algorithms [5]. As IP traffic matrix is typically not known in networks, these approaches have also been combined with traffic matrix estimation techniques [6], [7]. Traffic matrix estimation in [6] reroutes flows to estimate the traffic matrix, while the approach in [7] uses the gravity model for the same.

The existing network engineering techniques either minimize the cost of the capacity added to the network or minimize the number of new links added to the network. However, they do not consider the changes in the IP network topology which can lead to significant instabilities. While the aforementioned solutions can compute the new IP routing, and maybe even deployed with minimum service disruptions, they also lead to significant re-configurations in the network management systems of these networks. Rerouting of flows can disrupt the monitoring systems inside the network which need to be reconfigured. Also, the event correlation database extensively used by network management systems for alarm suppression, failure prediction and detection requires extensive and tedious reconfigurations due to rerouting, which is why current network operators prefer overprovisioning as it guarantees routing and operation stability.

Our main contribution is in designing a mechanism that can ensure IP routing stability, while affecting a minimum number of IP flows in course of dynamic circuit setup in the IP layer. We show that our approach is superior to the widely spread, simplest way to restrict routing changes in IP networks which is to add capacity on existing critical links. Although the latter operation ensures that IP routing is not affected and therefore minimizes the configuration effort in the network management system, this approach is not efficient in terms of cost or number of new circuits added. In fact, we show in this paper that the total number and capacity of bypasses (links) is less than the total number of congested links in a network. It is important to emphasize that our proposal is targeted to handle frequent short term traffic churns and therefore gives more weightage to routing stability as compared to congestion due to long term effects which focus on resource optimization. Although we have presented a preliminary bypass computation method and the main objectives in our past paper in [8], we here majorly extend the model to take into consideration the challenging case where the IP traffic matrix is unknown, as well as to consider the routing constraints of bypasses in the circuit-switched (optical) layer.

III. BYPASS BASED ROUTING WITH TRAFFIC MATRIX

We present an Integer Linear Program (ILP) which can be used to compute the optimal set of bypasses when the traffic matrix is known. The bypasses are used to alleviate the congestion, which is detected after the threshold of maximum link utilization is exceeded. The physical topology graph $G^p(V^p, E^p)$, with vertices $v_i^p \in V^p$ and edges $e_{ij}^p \in E^p$, and the logical topology $G^l(V^l, E^l)$, with vertices $v_i^l \in V^l$ and edges $e_{ij}^l \in E^l$ are given. The logical topology represents the interconnection of routers in the IP layer as shown in Fig. 1, while the physical topology represents the transport network (e.g, optical, Ethernet). Without loss of generality, each node (router) in the logical graph is connected to a single unique node (switch) in the physical graph and no two nodes are assumed to be connected to the same physical node. Also, for the sake of simplicity of the formulation, the nodes are numbered so that v_i^l in the logical topology is connected to the node v_i^p in the physical topology. The following information about the network is also assumed to be known:

- C_{ij}^p : The available capacity in the physical link $e_{ij}^p \in E^p$.
- C_{ij}^l : The total capacity of the logical link $e_{ij}^l \in E^l$.
- λ_{sd} : The traffic from a source v_s^l to destination v_d^l .
- α : Maximum acceptable utilization of a logical link.
- P_{ij} : Cost per unit bandwidth for e_{ij}^p .

We use T different bypass granularities, with bypass capacity C_{BP}^t and (interface) cost as $Cost^t$. The total cost of setting up a bypass is the sum of the cost of the link capacity used in the physical layer and the transmitter/receiver interface cost.

We assume that the IP routing of flows in the network is known, as it is usually the case, and that it does not change during the optimization process. We also assume that a flow from a source to a destination uses a single path in this formulation, and introduce two routing parameters:

- ψ_{xy}^{sd} : Boolean to indicate if the traffic from v_s^l to v_d^l uses the loose path $v_s^l \rightarrow v_x^l \rightarrow v_y^l \rightarrow v_d^l$. Note that $x \neq y$.
- $\psi_{xy}^{sd}(ij)$: Boolean to indicate if the traffic from v_s^l to v_d^l uses the loose path $v_s^l \rightarrow v_x^l \rightarrow v_i^l \rightarrow v_j^l \rightarrow v_y^l \rightarrow v_d^l$, given that $e_{ij}^l \in E^l$.

The first parameter ψ_{xy}^{sd} indicates if the route from v_s^l to v_d^l traverses over v_x^l and v_y^l , and therefore if a bypass from v_x^l to v_y^l can be used to re-route traffic from v_s^l to v_d^l . The parameter $\psi_{xy}^{sd}(ij)$ indicates if the link e_{ij}^l would be bypassed by a bypass from v_x^l to v_y^l for the flow from v_s^l to v_d^l . For example, in Fig. 2(a), a bypass from Z to D is a valid bypass for the path from X to D , indicating that $\psi_{zd}^{xd} = 1$ and the bypass traverses link $Z - Y$ indicating that $\psi_{zd}^{xd}(zy) = 1$.

We use three variables to identify the position of the bypass in the logical layer and the routing of this bypass in the physical layer, namely:

- X_{xy}^t : Boolean to indicate if a bypass of type t exists from node v_x^l to v_y^l .
- f_{xy}^{sd} : Boolean to indicate if the traffic from v_s^l to v_d^l is bypassed over a bypass from v_x^l to v_y^l .
- $r_{xy}^t(ij)$: Boolean to indicate if the bypass X_{xy}^t uses the link e_{ij}^p in the physical layer.

Note that the variables X_{xy}^t and f_{xy}^{sd} indicate the existence of the bypass in the IP layer and the routing of flows on this bypass. However, the bypass itself must be routed in the physical layer and this routing is defined by the variable $r_{xy}^t(ij)$. We define the objective function in Eq. 1 which minimizes the cost of setting up the bypasses. The first sum indicates the cost of interfaces used while the second sum is the cost of transport capacity required in the physical layer to set up the bypasses.

$$\text{Min} : \sum_t \text{Cost}^t \sum_{xy} X_{xy}^t + \sum_t C_{BP}^t \sum_{ij} P_{ij} \sum_{xy} r_{xy}^t(ij) \quad (1)$$

Subject to Constraints:

$$\forall v_s^l, v_d^l, v_x^l, v_y^l \in V^l : f_{xy}^{sd} \leq \psi_{xy}^{sd} \quad (2)$$

$$\forall v_s^l, v_d^l \in V^l, e_{ij}^l \in E^l : \sum_{xy} \psi_{xy}^{sd}(ij) \cdot f_{xy}^{sd} \leq 1 \quad (3)$$

$$\forall v_s^l, v_d^l, v_x^l, v_y^l \in V^l : f_{xy}^{sd} \leq \sum_t X_{xy}^t \quad (4)$$

$$\forall v_x^l, v_y^l \in V^l : \sum_{sd} \lambda_{sd} \cdot f_{xy}^{sd} \leq \alpha \sum_t X_{xy}^t \cdot C_{BP}^t \quad (5)$$

$$\forall e_{ij}^l \in E^l : \sum_{sd} \lambda_{sd} \cdot \psi_{ij}^{sd} \left(1 - \sum_{xy} \psi_{xy}^{sd}(ij) \cdot f_{xy}^{sd} \right) \leq \alpha C_{ij}^l \quad (6)$$

$$\forall v_x^l, v_y^l \in V^l : \sum_t X_{xy}^t \leq 1 \quad (7)$$

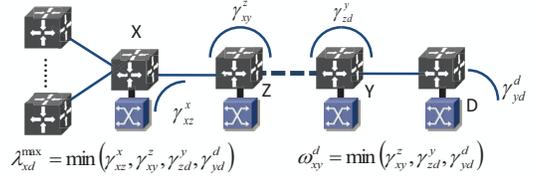
$$\forall t \in T, v_x^p, v_y^p \in V^p, : \sum_i r_{xy}^t(xi) = X_{xy}^t \quad (8)$$

$$\forall t \in T, v_x^p, v_y^p \in V^p, : \sum_i r_{xy}^t(iy) = X_{xy}^t \quad (9)$$

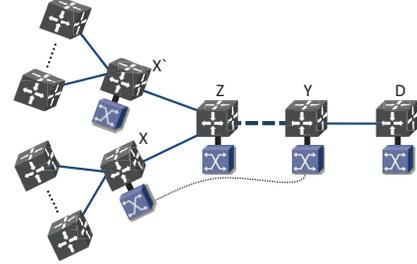
$$\forall t \in T, v_x^p, v_y^p, v_i^p \in V^p, i \neq x, y : \sum_k r_{xy}^t(ki) = \sum_j r_{xy}^t(ij) \quad (10)$$

$$\forall e_{ij}^p \in E^p : \sum_t \left(C_{BP}^t \sum_{xy} r_{xy}^t(ij) \right) \leq C_{ij}^p \quad (11)$$

Eq. 2, 3 and 4 define the constraints for routing of flows over bypasses in the logical layer. Eq. 2 indicates that traffic from v_s^l to v_d^l may be routed on a bypass from v_x^l to v_y^l iff the original route goes via nodes v_x^l and v_y^l . This constraint ensures that links not on the original path of a traffic flow are not affected by the bypassing of the said flow. The constraint in Eq. 3 ensures that different bypasses chosen to reroute a given flow do not overlap. In case of an overlap of bypasses selected for the same flow, there will be at least one link e_{ij}^l which is bypassed by multiple bypasses for the same flow implying that $\sum_{xy} \psi_{xy}^{sd}(ij) \cdot f_{xy}^{sd} > 1$, and hence would violate the constraints. For instance, in Fig. 2(a) bypasses from X to Y and from Z to D cannot be simultaneously used to bypass a flow from X to D as they both bypass the link $Z - Y$. Finally, the constraint in Eq. 4 ensures that a flow can only be bypassed from v_x^l to v_y^l if a bypass exists between these nodes and Eq. 5 constrains the capacity used by flows rerouted on the bypass. Eq. 6 defines the link capacity constraint for



(a) Computation of max bounds of individual and aggregate traffic flows



(b) Residual traffic on a link after rerouting traffic to specific destinations
Fig. 2. Illustrative examples used in ILP based models

the logical links in the network. The term $\lambda_{sd} \cdot \psi_{ij}^{sd}$ is used to determine if a flow from v_s^l to v_d^l uses the link e_{ij}^l , while the term $\left[1 - \sum_{xy} \psi_{xy}^{sd}(ij) \cdot f_{xy}^{sd} \right]$ is used to determine if the specified flow is bypassed over the link. If the flow is bypassed, the sum $\sum_{xy} \psi_{xy}^{sd}(ij) \cdot f_{xy}^{sd} = 1$, as constrained in Eq. 3, and the traffic for that particular flow is not taken into consideration.

Eq. 7 ensures that there is only one bypass established between any pair of nodes. This constraint is introduced primarily to reduce the complexity of the ILP, as if multiple bypasses were allowed between a pair of nodes, the flow assignment variable f_{xy}^{sd} would also be dependent on the class of bypass used to reroute the flow. We extend our primary ILP from [8] by introducing the constraints in Eq. 8, 9 and 10 to define the routing of the bypasses in the physical layer, with Eq. 8 and 9 ensuring that if a bypass of type t from node v_x^p to v_y^p exists, at least one outgoing link at the node v_x^p and one incoming link at the node v_y^p is used to route the bypass. Note here that v_x^l, v_y^l are connected to v_x^p and v_y^p respectively. Eq. 10 ensures that routing continuity of a bypass in the physical layer, and Eq. 11 constrains the capacity used on a physical link by different bypasses routed in the physical layer. The physical path variable $r_{xy}^t(ij)$ is also used in the objective function (Eq. 1) to compute the circuit bandwidth cost in the physical layer.

IV. BYPASS BASED ROUTING WITHOUT TRAFFIC MATRIX

The ILP presented in the previous section can compute the optimal set of bypasses for a given traffic matrix. However, finding the exact traffic matrix in the IP layer is not trivial, and must be estimated by alternate means. In this section, we present an ILP based model which does not assume knowledge of the traffic matrix and uses alternate metrics easily measurable in the commercial IP routers to compute the set of bypasses required to alleviate congestion. Information about the IP routing and link loads in the network can be used to compute maximum bounds on all λ_{sd} values, with λ_{sd}^{max} bounded by the minimum traffic on a link along the routing

path from v_s^l to v_d^l . A more accurate bound for maximum λ_{sd} values can be estimated by observing the virtual output queues inside IP routers. At any router v_x^l on the routing path from v_s^l to v_d^l , for an incoming link e_{ix} such that $\psi_{ix}^{sd} = 1$ and an outgoing link e_{xj} such that $\psi_{xj}^{sd} = 1$, the traffic λ_{sd} should be less than or equal to the total traffic routed from link e_{ix} to link e_{xj} . We define a new parameter γ_{df}^e as the traffic on the link e_{de}^l routed to link e_{ef}^l at node v_e^l . The index of the superscript indicates the router where the measurement is made while the subscript indicates the upstream and downstream router indices respectively. Special cases of this parameter include the traffic inserted at a node v_e^l on an link e_{ef}^l indicated by γ_{ef}^e and the traffic destined for node v_e^l on an link e_{de}^l indicated by γ_{de}^e .

Aggregate Flows versus Individual Flows: As shown in Fig. 2(a), the parameter γ_{df}^e can be used to compute the max traffic bound for traffic between a source-destination pair as shown below in Eq. 12. The maximum traffic bound is computed as the minimum of the traffic injected at the source link which is part of the route towards the destination, the traffic emerging at the destination from the link along the routing path from source to the destination, and traffic forwarded between two consecutive links along the routing path from the source to the destination.

$$\lambda_{sd}^{max} = \min \left\{ \begin{array}{l} (\gamma_{xz}^y : y \neq s, d, \psi_{xy}^{sd} = \psi_{yz}^{sd} = 1) \\ (\gamma_{sx}^s : \psi_{sx}^{sd} = 1) \\ (\gamma_{yd}^d : \psi_{yd}^{sd} = 1) \end{array} \right\} \quad (12)$$

Note that if the term γ_{sx}^s , (the traffic injected at the source), was not taken into consideration, the expression in Eq. 12 would provide a bound for all traffic upstream and from the node v_s^l to the destination v_d^l , when the routing path of all upstream traffic flows passing from v_s^l to v_d^l is same as the routing path from v_s^l to v_d^l . As is typically the case in current IP networks, routing decisions are based purely on IP addresses and typically follow a single path mechanism. Therefore, as illustrated in Fig. 2(a), a new parameter ω_{pq}^d is introduced which computes the total flow to destination v_d^l that can be bypassed v_p^l to v_q^l and is given by:

$$\omega_{pq}^d = \psi_{pq}^{pd} \cdot \min \left\{ \begin{array}{l} (\gamma_{xz}^y : y \neq p, d, \psi_{xy}^{pd} = \psi_{yz}^{pd} = 1) \\ (\gamma_{yd}^d : \psi_{yd}^{pd} = 1) \end{array} \right\} \quad (13)$$

As shown in the example in Fig. 2(a), for all flows from sources upstream and at X to D , if the max bound is determined by the traffic forwarded between consecutive links between X and D , the max bound for individual flows would be the same as the max bound for aggregate flows. Therefore, it is easy to recognize that the max bound for aggregate flows is at least equal, but likely tighter than the sum of individual max bounds in a very large number of cases. Clearly, the bound for aggregate traffic flows from or upstream of node v_s^l to v_d^l is then a better option than the bound on individual source-destination flows. We modify our bypass mechanism and instead of rerouting individual source-destination flows, we reroute aggregate flows to the destination at the ingress of

a bypass, with the same condition that the bypass egress is at the destination or is upstream from the destination.

Algorithm 1: Algorithm to compute $\hat{i}_{xy}^d(ij)$

```

 $\hat{i}_{xy}^d(ij) = 0$ 
if ( $\psi_{xy}^{xd}(ij) == 0$ ) then
  |  $\hat{i}_{xy}^d(ij) = \omega_{ij}^d$ 
else
  | if ( $x == i$ ) then
  | |  $\hat{i}_{xy}^d(ij) = 0$ 
  | else
  | | for ( $p$  s.t.  $\psi_{xi}^{xd}(pq) == 1$  or  $\psi_{xi}^{xd}(qp) == 1$  for
  | | some  $q, p \neq x$ ) do
  | | | for ( $r$  s.t.  $e_{rp}^l \in E^l, \psi_{xi}^{xd}(rp) == 0$ ) do
  | | | |  $\hat{i}_{xy}^d(ij) += \omega_{rp}^d$ 
  | | | |  $\hat{i}_{xy}^d(ij) += \lambda_{pd}^{max} \cdot \psi_{pq}^{pd}$ 
  | | | if ( $\hat{i}_{xy}^d(ij) > \omega_{ij}^d$ ) then
  | | | |  $\hat{i}_{xy}^d(ij) = \omega_{ij}^d$ 

```

Residual Flows after Bypass: As described above, we now bypass all traffic to a destination from the ingress of the bypass. Now consider the scenario as shown in Fig. 2(b): a bypass is established from router X to Y towards destination D . While the bypass reroutes all traffic from and upstream of X to D across the link $Z - Y$, the link $Z - Y$ still has traffic for D which is inserted at and upstream from X and the traffic inserted at Z for D . In order to compute the traffic on a link after flows upstream of the link have been bypassed, we introduce a new parameter $\hat{i}_{xy}^d(ij)$ which estimates the *maximum residual traffic bound* on the link e_{ij}^l to the destination v_d^l , if the bypass from v_x^l to v_y^l is used to reroute all traffic to v_d^l .

The parameter $\hat{i}_{xy}^d(ij)$ can be computed as the traffic inserted by all nodes downstream from v_x^l till v_i^l to v_d^l as well as the traffic for flows which have one or more of the nodes downstream from v_x^l till v_i^l but not the node v_x^l in their routing paths. For example, in Fig. 2(b), the max bound for traffic to D traversing the link $Z - Y$ can be computed as ω_{XZ}^D (traffic from and upstream of X) + λ_{ZD}^{max} (traffic inserted at the node Z for destination D). We use Algo. 1 to compute $\hat{i}_{xy}^d(ij)$. The algorithm first checks if rerouting of traffic to destination v_d^l over the bypass from v_x^l to v_y^l affects any flows on the link e_{ij}^l , and if not, the max bound of traffic to v_d^l on the link remains unchanged. On the other hand, if the bypass originates at the ingress node v_i^l ($x == i$), all traffic to destination v_d^l is bypassed and therefore traffic to v_d^l on link e_{ij}^l is equal to 0. For the scenario shown in Fig. 2(b), the algorithm traverses all nodes along the path from v_x^l to v_z^l (v_x^l not included), and includes the max bounds for the traffic inserted at these nodes (λ_{zd}^{max}) and the traffic from all neighboring nodes connected to these nodes. Note here that the sum of max bounds is limited

by the max. bound of total traffic on link $Z - Y$ towards D given by ω_{ZY}^d .

We now present the ILP formulation for the computation of bypasses. As all flows to a particular destination are bypassed now, the flow indicator variable f_{xy}^{sd} is replaced by f_{xy}^d , which indicates if the flow to destination v_d^l is bypassed over the bypass from v_x^l to v_y^l . The objective function and the physical routing constraints remain unchanged in this formulation as they only depend on the variable X_{xy}^t which determines if a bypass is required between the nodes v_x^p and v_y^p .

$$\text{Min} : \sum_t \text{Cost}^t \sum_{xy} X_{xy}^t + \sum_t C_{BP}^t \sum_{ij} P_{ij} \sum_{xy} r_{xy}^t(ij) \quad (14)$$

Subject to Constraints:

$$\forall v_d^l, v_x^l, v_y^l \in V^l : f_{xy}^d \leq \psi_{xy}^{xd} \quad (15)$$

$$\forall v_d^l \in V^l, e_{ij}^l \in E^l : \sum_{xy} \psi_{xy}^{xd}(ij) \cdot f_{xy}^d \leq 1 \quad (16)$$

$$\forall v_d^l, v_x^l, v_y^l \in V^l : f_{xy}^d \leq \sum_t X_{xy}^t \quad (17)$$

$$\forall v_x^l, v_y^l \in V^l : \sum_d \omega_{xy}^d \cdot f_{xy}^d \leq \alpha \sum_t X_{xy}^t \cdot C_{BP}^t \quad (18)$$

$$\forall e_{ij}^l \in E^l : \sum_d \omega_{ij}^d \cdot \left(1 - \sum_{xy} \psi_{xy}^{xd}(ij) \cdot f_{xy}^d \right) + \sum_d \sum_{xy} f_{xy}^d \cdot \hat{i}_{xy}^d(ij) \leq \alpha C_{ij}^l \quad (19)$$

$$\forall v_x^l, v_y^l \in V^l : \sum_t X_{xy}^t \leq 1 \quad (20)$$

$$\forall t \in T, v_x^p, v_y^p \in V^p, : \sum_i r_{xy}^t(xi) = X_{xy}^t \quad (21)$$

$$\forall t \in T, v_x^p, v_y^p \in V^p, : \sum_i r_{xy}^t(iy) = X_{xy}^t \quad (22)$$

$$\forall t \in T, v_x^p, v_y^p, v_i^p \in V^p, i \neq x, y : \sum_k r_{xy}^t(ki) = \sum_j r_{xy}^t(ij) \quad (23)$$

$$\forall e_{ij}^p \in E^p : \sum_t \left(C_{BP}^t \sum_{xy} r_{xy}^t(ij) \right) \leq C_{ij}^p \quad (24)$$

The routing constraints in the IP layer are presented in Eq. 15, 16 and 17. Eq. 15 ensures that the node v_y^l is on the original routing path from v_x^l to v_d^l . Eq. 16 ensures that traffic to v_d^l is not bypassed by overlapping bypasses. Unlike the previous ILP, overlapping bypasses do not lead to routing misconfigurations here. For instance, in Fig. 2(b), a bypass from X to Y and a bypass from Z to Y can both be used to bypass traffic to the destination D . However, given that we only have max. bound information for traffic bypassed at X and Z , the estimate of residual traffic is very lax leading to inefficient solutions. The third routing constraint (Eq. 17) ensures that flows can only be bypassed from v_x^l to v_y^l if a bypass exists between the nodes. The bypass capacity constraint in Eq. 18 uses the max. bound of traffic from and

upstream of the ingress node v_x^l to the destination v_d^l (ω_{xy}^d) to evaluate the required bypass capacity. In the link capacity constraint (Eq. 19) traffic on a link is expressed as a sum of max bounds of traffic to all destinations: If there is no bypass, $\left(\left(1 - \sum_{xy} \psi_{xy}^{xd}(ij) \cdot f_{xy}^d \right) = 1 \right)$, the max. bound of all traffic to destination v_d^l i.e. (ω_{ij}^d) is used. If there is a bypass from v_x^l to v_y^l , the max. bound on the residual traffic to the destination v_d^l i.e. $\hat{i}_{xy}^d(ij)$ is used. As max. traffic bounds are used, estimated traffic on a link may be more than the actual traffic, and hence in our implementation, this constraint is only applied to congested links which are known apriori, as load on non-congested links cannot be increased by bypass establishment.

V. NUMERICAL RESULTS

We study the performance of the proposed ILPs on the Atlanta reference network using the traffic matrix given in [9]. The Atlanta network is a mesh based 15 node network with 22 links and average nodal degree of 2.93. The physical and the logical layer topologies are assumed to be the same, with each link in the physical topology having a normalized cost per unit bandwidth = 1, and the total available capacity = 100,000. The shortest path first (SPF) routing is used in the IP layer, and the link capacities are assigned so that the initial link utilization of all links is 0.71. We use nine different bypass types with capacity and normalized interface costs as shown in Fig. 3. To test the performance of our scheme, we randomly select a number of (s, d) pairs and increase the traffic between them by 150%. We then use the ILP to compute the optimal bypass set for congested links (utilization $> \alpha = 0.9$). For each result set, the values are averaged over 20 runs.

Capacity	Cost	Capacity	Cost	Capacity	Cost
100	200	1,000	500	10,000	800
200	300	2,000	600	20,000	900
500	400	5,000	700	50,000	1000

Fig. 3. Bypass types and costs

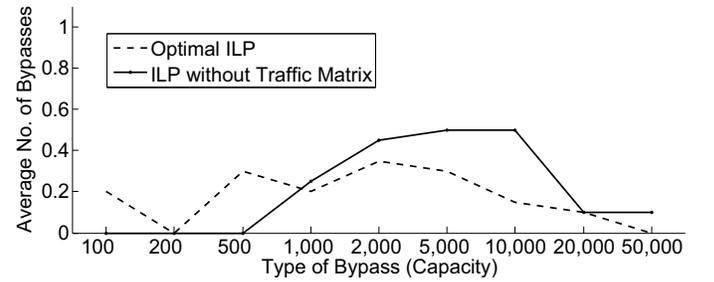


Fig. 4. Average number of bypasses computed by the ILPs

Fig. 4 depicts the average number of bypasses computed by both the ILPs when 7 to 8 flows observe an increase in traffic. It can be seen that the knowledge of the traffic matrix and the ability to switch individual source-destination flows instead of aggregate flows allows the first ILP to create more bypasses with smaller granularity. The second ILP tends to

create bypasses with higher granularity due to the inaccurate max. bounds used to estimate the flows on the bypass. The inaccuracy of the max. bound is also the reason of the low bypass utilization as shown in Fig. 5, where the average utilization of bypass capacity is almost half of the utilization measured in the optimal scenario with known traffic matrix.

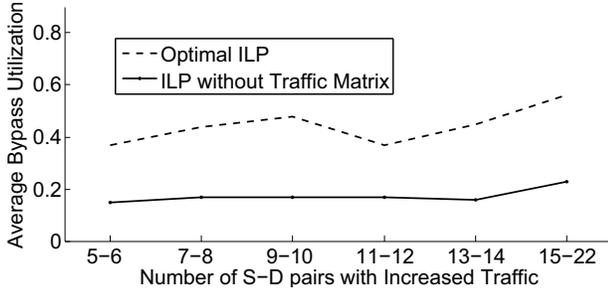


Fig. 5. Number of bypasses of different classes computed by the ILPs

The total number of bypasses required by both solutions are shown in Fig. 6. While the bypass capacity utilization is very poor, on an average, the total number of bypasses required by the sub-optimal solution is not significantly higher than that for the optimal ILP. Fig. 6 also shows that the total number of bypasses established on average for both solutions is lower than the number of congested links, suggesting that the proposed mechanism is much better suited than simply installing additional capacity.

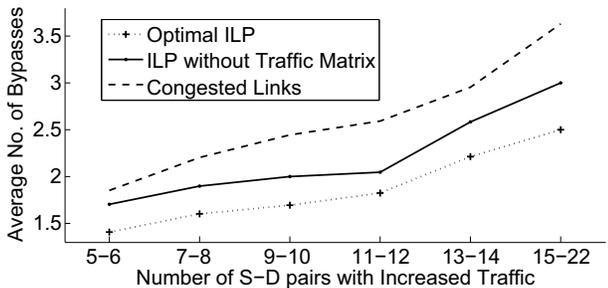


Fig. 6. No. of bypasses for different no. of flows with increased traffic

Finally, the result in Fig. 7 shows the number of IP flows that are bypassed. The optimal ILP only bypasses approximately 1 percent of the flows when 7-10 percent of the flows observe an increase in traffic (total flows = 210). The second ILP uses insufficient traffic information and reroutes all flows to a destination, and hence the number of destinations as well as the total number of $s - d$ flows bypassed are higher. While significantly larger than the values for the optimal ILP, the bypasses in this example still reroute only about 5 percent of flows when 7-10 percent of flows observe increase in traffic.

It can be seen that when the traffic matrix is not known, the ILP solution tends to significantly over-provision the capacity of the dynamic circuits. However, as seen in Fig. 6, the number of bypasses introduced by this solution are not much higher than the optimal solution, with the difference approximately only 0.3 bypasses. This indicates that if our solution is coupled with transport technologies which have the capability to flexibly change the capacity of a circuit, such

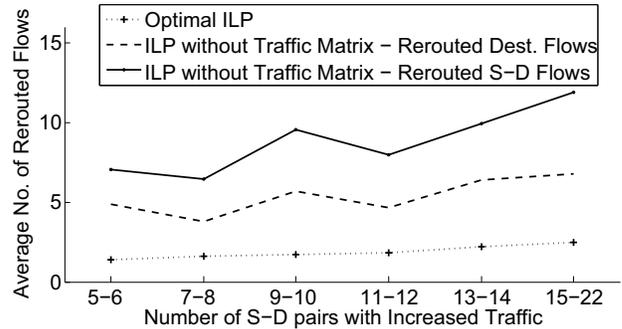


Fig. 7. Average number of source destination flows rerouted by ILPs

as with carrier-grade Ethernet, we can observe the traffic on the bypass after the network has settled, and then modify the capacity of the established circuits to match the actual bypassed traffic load. Also, as seen in Fig. 7, the total number of flows rerouted via the bypasses are small indicating that IP routing is minimally affected.

VI. CONCLUSIONS

We presented a new network engineering paradigm which deploys dynamic optical bypass to alleviate congestion in the IP layer while keeping the IP routing stable. Our results show that the number of bypassed flows is relatively small, indicating that network operators are not subject to heavy reconfigurations during bypass establishment and teardown. The results show that even without the knowledge of the traffic matrix, the proposed ILP can compute bypasses, albeit at a higher cost due to the typically higher capacity of the resulting bypasses. However, these results are still promising as the total number of bypasses generated without the knowledge of the traffic matrix is only marginally higher and when applied in conjunction with technologies where circuit capacity can be changed on demand, such as in carrier-grade Ethernet, the overall performance is expected to be close to optimal. As the next step, it is necessary to develop efficient heuristics for the same as the ILPs optimizers cannot be deployed in real time in response to a short-lived congestion.

REFERENCES

- [1] Internet2 Headroom Practice, <https://wiki.Internet2.edu/confluence/download/attachments/17383/Internet2+Headroom+Practice+8-14-08.pdf>
- [2] L. Cheng, et al. "Network Engineering-Control of Dynamic Link Topology in User Networks," Bell Labs Technical Journal, 8(1),207-218
- [3] S.F. Gieselmann, et al. "Minimum-cost virtual-topology adaptation for optical WDM mesh networks," IEEE ICC, May 2005
- [4] B. Mukherjee, et al. "Some Principles for Designing a Wide-Area WDM Optical Network," IEEE/ACM Transactions on Networking, vol. 4, no. 5, pp. 684-696, Oct. 1996.
- [5] R.J. Durn, et al. "Genetic algorithm to design logical topologies in reconfigurable WDM networks," Photonic Network Communications, vol. 17, no. 1, pp. 21-33, Feb. 2009
- [6] Y. Ohsita, et al. "Gradually Reconfiguring Virtual Network Topologies based on Estimated Traffic Matrices," IEEE INFOCOM, May 2007
- [7] Y. Zhang, et al. "Fast Accurate Computation of Large-Scale IP Traffic Matrices from Link Loads," ACM SIGMETRICS, June 2003
- [8] M. Chamania, M. Caria, A. Jukan, "Achieving IP Routing Stability with Optical Bypass," IEEE ANTS 2009, Dec 2009.
- [9] Atlanta reference Network, <http://sndlib.zib.de/coredata.download.action?objectName=atlanta&format=native&objectType=network>