Making Intra-Domain Routing Robust to Changing and Uncertain Traffic Demands: Understanding Fundamental Tradeoffs

David Applegate
AT&T Labs–Research
180 Park Avenue
Florham Park, NJ 07932, USA
david@research.att.com

Edith Cohen
AT&T Labs–Research
180 Park Avenue
Florham Park, NJ 07932, USA
edith@research.att.com

ABSTRACT

Intra-domain traffic engineering can significantly enhance the performance of large IP backbone networks. Two important components of traffic engineering are understanding the traffic demands and configuring the routing protocols. These two components are inter-linked, as it is widely believed that an accurate view of traffic is important for optimizing the configuration of routing protocols and through that, the utilization of the network.

This basic premise, however, never seems to have been quantified – How important is accurate knowledge of traffic demands for obtaining good utilization of the network? Since traffic demand values are dynamic and illusive, is it possible to obtain a routing that is "robust" to variations in demands? Armed with enhanced recent algorithmic tools we explore these questions on a diverse collection of ISP networks. We arrive at a surprising conclusion: it is possible to obtain a robust routing that guarantees a nearly optimal utilization with a fairly limited knowledge of the applicable traffic demands.

Categories and Subject Descriptors

C.2 [**Communication Networks**]: C.2.2 Network Protocols;C.2.3 Network Operations; F.2 [**Analysis of Algorithms**]: F.2.2 Nonumerical Algorithms and Problems

General Terms

Algorithms,Design,Management,Performance,Reliability

Keywords

routing; TM estimation; demand-oblivious routing

1. INTRODUCTION

Intra-domain traffic engineering has gained a lot of popularity in the recent years – good traffic engineering tools can significantly contribute to the management and performance of large operational IP networks [21, 2]. Two important components of traffic engineering are understanding traffic flows, and configuring (and designing) routing protocols. These two components are related – it is widely accepted that good understanding of the traffic matrix (TM) and the dynamics of traffic flows can lead to better utilization of link capacities through more appropriate routing of traffic [9]. Theoretically, if the TM is known exactly, then an optimal routing for it can be obtained by solving the corresponding multi-commodity flow problem instance [16]; and with OSPF/IS-IS, the most common intra-domain routing protocol, link weights can be tuned according to the TM to often yield near-optimal utilization [10].

Unfortunately, measuring and predicting traffic demands are illusive problems [21, 2]. Flow measurements are rarely available on all links and Egress/Ingress points of the network, and it is even harder to estimate Origin-Destination flow aggregates. Moreover, demands change over time – on a diurnal cycle and less predictably as a result of special events or failures internal or external to the network. These problems were recently tackled with models and measurement tools [5, 9, 8, 15, 19] that allow one to extrapolate and estimate traffic demands. It seems, however, that the most one can hope for is some approximate picture of demands, and not necessarily even a very current one.

Even if current demands are known, their dynamic nature poses a challenge: On one hand, it is desirable for the routing to be efficient on the current traffic demands, thus, to be adjusted as demands shift. On the other hand, one would like to limit modifications to the routing, since changes can potentially cause disruptions in service due to path shifts and convergence time while the system reaches a consistent state. For OSPF/IS-IS routing, this tradeoff was explored in [11], which developed a technique that limits the amount of change to the OSPF/IS-IS link weights (which determine the routings) when the TM changes.

Good system engineering thus calls for a design that it robust under a range of conditions. That is, a routing that can perform nearly optimally for a wide range of applicable traffic demands. Our primary goal is to explore the viability of such a routing, that is, to understand the sensitivity of the quality of attainable routing to the extent within which we know the traffic demands. While both these basic traffic engineering building blocks, routing and TM estimation,

SIGCOMM'03, August 25–29, 2003, Karlsruhe, Germany.
Copyright 2003 ACM 1-58113-735-4/03/0008 ...$5.00.

are intensively studied, their interaction, and its underlying performance tradeoffs, are not well understood.

Although it is widely believed that understanding traffic demands is necessary for achieving good utilization of the network [21, 2, 9, 15], this belief was never carefully quantified: How well can a routing designed with no knowledge (or only ball-park knowledge) of the TM perform? That is, how precise an estimate of traffic demands is needed in order to guarantee good utilization? When traffic demands shift, what range of change is tolerable within some performance guarantees? How would a routing designed to be optimal for a specific TM perform when the actual traffic demands deviate from the presumed ones?

Lastly, we consider the performance of different routings in the event of link failures. When failures occur, the optimal routing strategy can be recomputed from scratch, resulting in optimal performance ratio but possibly in large shifts in flow patterns. (This tradeoff between utilization and traffic shifts, in the context of OSPF routing, is investigated in [11].) We thus compare the performance of the existing routing on the "failed" network (where only traffic flows which traverse the failed link are shifted), to the best possible routing on the failed network.

The questions we raise concern fundamental limits and tradeoffs for managed IP networks – we expect these issues to remain relevant as routing protocols evolve – in particular with deployment of more sophisticated tuning of OSPF/IS-IS weights [10, 11], and with the gradual deployment of more flexible protocols such as Multi Protocol Label Switching (MPLS) protocol [1, 18] and its future successors.

The pursuit of answers to these questions requires a way to measure how well a given routing performs on a range of traffic demands and a way to design a routing which performs well on an appropriately wide range of traffic demands. But while previously known algorithms can obtain an optimal routing for a specific TM (or a small set of TMs), they can not be extended to work on a wide range of TMs. At the heart of our work are novel algorithms, based on which we built software for producing an optimal routing for a range of possible TMs. This routing optimally balances the load across the range of TMs – it minimizes the extent to which the maximum link utilization of any TM deviates from the best possible by the optimal routing that is tailored for that TM. Our software also enabled us to compare different routings by computing the worst performance ratio obtained by each routing on the range of applicable TMs.

Our evaluation utilizes maps of a diverse collection of ISPs, made available by the Rocketfuel project [20, 14], and the test network studied in [15]. The data is described in Section 2 and our performance metrics and methodology are described in Section 3 followed by evaluation results in Section 4. The LP models we used are developed in Section 5. Our evaluation is complemented with some asymptotic analysis on some simple network structures presented in Section 6.

2. DATA

We describe the test topologies we used. Unfortunately, ISPs regard their topologies as proprietary information, and until recently, researchers had to settle for proprietary information synthetic data; conclusions thus often suffered from a lack of generality and verifiability. A recent breakthrough was made by the Rocketfuel project [20], which developed a new set of measurement techniques and released publicly-available approximate router-level topologies of a diverse and representative collection of ISPs. We used heuristics to augment this data with link capacities and traffic matrices.

2.1 Topologies

We use the six ISP maps from the Rocketfuel dataset which had accompanying (deduced) OSPF/IS-IS weights [14]. We then collapsed the topologies so that "nodes" correspond to cities to obtain approximate PoP to PoP (Point of Presence) topologies. We also included the 14-node and 25-link "Tier-1 PoP to PoP topology" evaluated in [15] (labeled as "N-14" in the sequel). The studied topologies are listed in Table 1.

2.2 Capacities

The topologies provided by Rocketfuel and in [15] did not include the capacities of the links, which were needed for our study. The Rocketfuel maps did include derived OSPF/IS-IS weights of links [14], which were computed to match observed routes. In the absence of any other information on capacities, we used the weights to associate hopefully compatible capacities by "turning around" the Cisco-recommended default setting of link weights according to capacities: The Cisco default setting for OSPF weights is to set the weight of each link to be inversely-proportional to its capacity [6].

2.3 Traffic matrices (TMs)

Accurate traffic matrices are not generally available. Not only are they regarded as proprietary by ISPs, but, as noted in the introduction, they are hard to obtain with reasonable accuracy. We thus used two families of synthetic traffic matrices, which we refer to as *Bimodal* and *Gravity* TMs:

Bimodal TMs. It was observed that only a fraction of Origin-Destination (OD) pairs has very large flows [4]. This model assumes that these flows dominate the points of congestion. The random bimodal distribution samples randomly a fraction of OD pairs and then assigns a demand for the pair uniformly at random from some range.[1] Random bimodal distributions (and other random distributions) were used in [15]).

Gravity TMs. Since networks are designed with some expectation of traffic demands in mind, it is desirable to evaluate the performance of different routings with respect to such traffic demands. We used a Gravity model, similarly to that suggested in [19], to generate demands that "correspond" to the network. The work in [19] suggested a way to extrapolate a complete TM from measurements of incoming-outgoing flow into each PoP from the backbone. The extrapolation then assumed that the fraction of traffic sourced from a PoP is sinked at other PoPs proportionally to the total sinked flow at these PoPs. According to [19] this simple model is surprisingly accurate. Since we did not have even these more restricted flow values, we used a capacity-based heuristic, which assumes that the incoming/outgoing flow from each PoP is proportional to combined capacity of connecting links. We then applied the gravity model as in [19] to extrapolate a complete TM.

[1] Distributions other than uniform or the particular parameter settings did not seem to make a qualitative difference in the results.

AS	routers	orig-links	cities	links	reduced cities	reduced links
Telstra (Australia) 1221	108	306	57	59	7	9
Sprintlink (US) 1239	315	1944	44	83	30	69
Ebone (Europe) 1755	87	322	23	38	18	33
Tiscali (Europe) 3257	161	656	50	88	28	66
Exodus (Europe) 3967	79	294	22	37	21	36
Abovenet (US) 6461	141	748	22	42	17	37
N-14 (MTSBD02)			14	25		

Table 1: Topologies from Rocketfuel (with AS number and name) and [15] (the N-14) network. The table lists the number of routers and links, the number of cities and inter-city links which we refer to as PoPs. The last two columns (reduced cities and links) list the number of remaining cities and links if 1-connected components ("hanging" trees) are removed. These components do not change the relative quality of different routings (see Lemma 5.1), thus we were able to perform some computations faster on these reduced graphs.

3. METRICS AND METHODOLOGY

3.1 Routing

A *routing* specifies how traffic of each Origin-Destination (OD) pair is routed across the network. Typically there is path diversity, that is, there are multiple paths for each OD pair, and each path routes a fraction of the traffic.

Open Shortest Path First (OSPF) or Intermediate System-Intermediate System (IS-IS) protocols specify a routing through a set of link weights. The traffic between each pair is always routed on shortest path(s) between the origin and destination (with respect to these weights). Typically, there are multiple shortest paths; when this happens, each router splits the outgoing traffic evenly on all applicable interfaces. By controlling the weights, many possible routings are possible. The Cisco-recommended default setting is to use link weights that are inversely proportional to the link capacities [6]. With more fine-tuned traffic engineering it is typically possible to select weights that are expected to work well on the projected TM [10]. The OSPF routing used in our evaluation is the routing obtained by the OSPF/IS-IS (estimated) link weights provided with our data. This routing should match reasonably closely the actual routing used by these ISPs [14].

The MPLS protocol allows for a rich (general) specification of routings and more fine tuned traffic engineering. Our optimization is with respect to routings of this more general form, that is, routing that can be implemented via MPLS but not necessarily via OSPF/IS-IS.

For our purposes, the relevant characterization of each routing is what fraction of traffic, for each OD pair, is routed along each link. Thus, the routing is specified by a set of values $f_{ab}(i,j)$ that specifies the fraction of demand from a to b that is routed on the link (i,j). Note that the values $f_{ab}(i,j)$ for a given OD pair $a \rightarrow b$, should specify a flow of value 1 from a to b. When the routing routes a demand d_{ab} for the OD pair $a \rightarrow b$, the contribution of this demand to the flow on a link (i,j) is $d_{ab}f_{ab}(i,j)$.

Our optimization algorithm generates an optimal routing with respect to a set of TMs. We next discuss our performance metrics for the "goodness" of a routing.

3.2 Metrics

A common metric for the performance of a given routing with respect to a certain TM is the *maximum link utilization*. This is the maximum, over all links, of the total flow on the link divided by the capacity of the link (see e.g. [10, 11]). Formally, the maximum link utilization of a routing $\mathbf{f}$ on TM $\mathbf{D}$ (where d_{ab} is the demand from a to b) is

$$\max_{(i,j)\in\text{links}} \sum_{a,b} d_{ab} f_{ab}(i,j)/\text{cap}_{ij} \ ,$$

where cap_{ij} is the capacity of the link (i,j).

An *optimal* routing for a certain TM $\mathbf{D}$ is a routing which minimizes the maximum link utilization. Formally, the optimal utilization for a TM $\mathbf{D}$ is given by

$$\text{OPTU}(\mathbf{D}) =$$

$$\min_{\mathbf{f}|\mathbf{f} \text{ is a routing}} \ \max_{(i,j)\in\text{links}} \frac{\sum_{a,b} d_{ab} f_{ab}(i,j)}{\text{cap}_{ij}} \ .$$

The *performance ratio* of a given routing $\mathbf{f}$ on a given TM $\mathbf{D}$ measures how far is $\mathbf{f}$ from being optimal on the TM $\mathbf{D}$. It is defined as the maximum link utilization of $\mathbf{f}$ on $\mathbf{D}$ divided by the minimum possible maximum link utilization on this TM. Formally,

$$\text{PERF}(\mathbf{f}, \{\mathbf{D}\}) = \frac{\max_{(i,j)\in\text{links}} \sum_{a,b} d_{ab} f_{ab}(i,j)/\text{cap}_{ij}}{\text{OPTU}(\mathbf{D})} \ .$$

Note that the performance ratio is always at least 1; it is exactly 1 if and only if the routing is optimal for $\mathbf{D}$.

It is well known that the optimal routing for a given TM can be computed by solving a corresponding multicommodity flow linear program (this routing was looked at in [16]). Note that this routing is optimized for a specific TM, thus, it does not provide performance guarantees for other TMs. This is important, since, as mentioned earlier, traffic patterns change over time and it is also not generally possible to obtain a good estimate of the current TM.

The definition of the performance ratio follows the "competitive analysis" framework where performance guarantees of a certain solution are provided relative to the best possible solution. We now extend the definition of performance ratio of a routing to be with respect to a set of TMs. Let D be a set of TMs. The performance ratio of a routing $\mathbf{f}$ on D is defined as

$$\text{PERF}(\mathbf{f}, D) = \max_{\mathbf{D}\in D} \text{PERF}(\mathbf{f}, \{\mathbf{D}\}) \ .$$

A routing $\mathbf{f}$ is optimal for the set D if and only if it minimizes the performance ratio, that is, $\text{PERF}(\mathbf{f}, D)$ is minimal. The performance ratio $\text{PERF}(\mathbf{f}, D)$ is always at least 1 – but note that the best possible performance ratio on the set of TMs D can be strictly larger than 1; since generally, a single routing that is optimal for all TMs in the set may not exist.

When the set D includes all possible TMs, we refer to the performance ratio as the *oblivious performance ratio* of a routing. The oblivious ratio is the worst performance ratio a routing obtains with respect to all TMs. A routing with a minimum oblivious ratio is an *optimal oblivious routing*, and its oblivious ratio is the *optimal oblivious ratio* of the network.

To better interpret the performance ratio, note that it is invariant under scaling of the TMs in the set D or of the link capacities. The performance ratio constitutes a comparative measure of *different routings, on a given topology and set of TMs*, but it is not a meaningful comparative measure between *different network topologies* – it is defined relative to the minimum possible maximum link utilization, but the min max utilization itself varies with topology. Also note that there can be many possible optimal routings and they can differ in how they perform on specific TMs. Illustrative examples and analysis of the optimal oblivious performance ratio on some simple networks are provided in Section 6.

3.3 Computing an optimal routing

Until recently, known tools allowed for optimizing the routing with respect to a given TM, but beyond specific highly structured topologies (such as hypercubic networks), not much was known about how to efficiently construct an optimal routing with respect to a broad set of demands and what are the optimal performance ratios. A recent breakthrough work by Räcke [17] showed (existentially) a surprising upper bound: all symmetric networks (that is, networks where link capacities are the same in both directions, as is typically the case with large backbone networks) have a routing with an oblivious ratio that is at most polylogarithmic in the number of nodes. Räcke's existential bound triggered the development of a *polynomial time* construction of an *optimal* oblivious routing [3] for *any network* (symmetric or not). The polynomial time algorithm in [3] is based on applying the Ellipsoid algorithm to an exponential-size LP model and as such is not practical for large networks. We develop a novel simpler and faster algorithm (both asymptotically and implementation wise) for computing an optimal oblivious routing that is based on a polynomial-size LP formulation (see details in Section 5). We then extend our model to optimize the routing with respect to range restrictions on OD-pair demands. In our simulations, we solve these LPs using the CPLEX LP solver [7] (other public-domain LP solvers could be applied as well).

3.4 Limitations

We conclude this section with discussion of limitations. Our models and metrics do not capture the interaction between traffic demands and the resulting actual throughput, we rather compare different routings through the maximum link utilization obtained if all demands are indeed routed. This is a reasonable metric as packet loss and congestion are more likely when the utilization is higher.

Our evaluation focuses on point to point (OD pair) demands rather than point-to-multipoint. Point-to-multipoint demands are often relevant to large ISPs (e.g. when there are multiple peering points to a different ISP and thus any of a number of egress points can be used interchangeably [9]). This point-to-point "restriction" stems mostly from the limitations of our data and in principle our techniques and software extend to cover point to multipoint demands.

Our optimizations are performed with respect to maximum link utilization and performance ratio. In specific implementation contexts our methodology can be augmented with other considerations (For example, when using MPLS, beyond capacity utilization one may want to optimize MPLS label stack size or the number of provisioned paths.).

	reduced	oblivious ratio:			time
ASN	pops/links	opt	OSPF	gravity-opt	(seconds)
1221	7/ 9	1.425	4.16	3.50	0.12
N-14	14/ 26	1.972	7.74	7.58	9.20
1755	18/ 33	1.781	16.60	8.15	30.58
6461	17/ 37	1.910	13.41	20.10	49.12
3967	21/ 36	1.623	49.20	12.92	51.13
3257	28/ 66	1.803	51.18	16.24	925.89
1239	30/ 69	1.895	233.98	31.57	1897.89

Table 2: Oblivious performance ratio on different topologies for the following routings: The optimal oblivious routing, the OSPF routing, and a routing which is optimal for Gravity TMs. The table lists the optimization time of computing the optimal oblivious routing on a Compaq Alphaserver ES40 with 500MHz processors and 4GB of memory.

4. EXPERIMENTS AND RESULTS

The first question we address is, what are the best performance ratio guarantees attainable on our test networks barring any knowledge of traffic demands? Table 2 lists the oblivious performance ratio for 3 different routings: The optimal oblivious routing (computed using the LP formulation in Section 5.3), and two other natural routings - the OSPF routing (using the weights provided in the dataset.[2].), and the optimal routing for the Gravity TMs (computed by solving a multi-commodity flow LP). The performance ratio of each given routing was computed using the "slave LP" formulation in Section 5.2. The optimal oblivious performance ratio on the evaluated topologies ranges from $1.425 - 1.972$, which means that these networks have a routing that on *any* TM is guaranteed to have maximum link utilization that is at most 43%-97% larger than that of the best possible routing that is tailored to this TM. The two other routings evaluated have significantly worse (2-3 digit) oblivious ratios, which means that on some TMs, they are very far from the tailored optimal routing. These gaps indicate that it is unlikely that an oblivious performance ratio that is close to optimal can be obtained in an ad hoc manner, without the use of our optimization tools.

A 43%-97% (worst case) overhead in max utilization is far from being negligible to working ISPs – the good news, however, is that such guarantees can be obtained with no knowledge whatsoever on the traffic demands.

[2] Recall that these weights were such that the derived OSPF routing is consistent with observed routes. Note that this OSPF routing is different from the *best OSPF-style routing*, that is, a set of link weights such that the corresponding OSPF routing has a minimum oblivious ratio. An independent interesting problem is to produce an optimal OSPF-style routing and compare its performance to the optimal MPLS-based routing on our test networks. The optimization, however, seems highly non-trivial as it can no longer modeled as an LP. Obviously, the OSPF-style optimal oblivious ratio is at least as large as the optimal (MPLS-style) oblivious ratio. Generally, the performance gap can be large (e.g., on clique networks), but one study [10] argues that "typically," *for a single TM,* the best OSPF routing nearly matches in performance the optimal MPLS routing.

Fortunately, however, even though an exact current estimate of the TM is typically very hard to obtain, much about the TM is known. The TM can vary within some known range or can be estimated to within some known accuracy. In this case, we would like a performance guarantee with respect to all TMs that lie within some range. The next question we examine is the sensitivity of the attainable performance ratio to the "error margin" within which the TM is known. (Note that as we expand the set of TMs with respect to which we compute the performance ratio, the ratio can only increase).

In this set of experiments we consider a topology, a TM (Bimodal or Gravity), and an error margin parameter $w \geq 1$. We consider a "base" TM, $\mathbf{D}$ (bimodal or gravity TM), which can be thought of as our best "guess" of the actual TM. The set of applicable TMs, D_w, includes each $\mathbf{D'}$ such that for all OD pairs (i, j), $d_{ij}/w \leq d'_{ij} \leq wd_{ij}$. This set can be thought of as including all TMs with respect to which we want a performance guarantee.

In our evaluation we compute the performance ratio of different routings on the set D_w (for different values of w.). The performance ratio for each given routing is computed with respect to all (infinitely many) TMs in the set D_w using the "slave LP" formulation given in Section 5.2 with the margin constraints added to it. The routings evaluated are:

- *opt*: An optimal routing for the range of demands D_w, that is, a routing which minimizes $\text{PERF}(\mathbf{f}, D_w)$. These routings are computed via our LP models developed in Section 5.4. (Note that there are potentially different routings for different values of w.)

- *no-margin-opt:* An optimal routing for the base TM (that is, a routing that minimizes $\text{PERF}(\mathbf{f}, \{\mathbf{D}\})$ or equivalently, the maximum link utilization when routing $\mathbf{D}$). This routing is computed by solving a multi-commodity flow LP.

- *OSPF:* The OSPF routing (using the weights provided in the dataset).

- *global-opt:* An optimal oblivious routing for the topology (that is, a routing that minimizes the worst-case performance ratio over all possible TMs). These routings are computed using the LP models developed in Section 5.3.

- *nm-gravity-opt:* When the base TM $\mathbf{D}$ is Bimodal, we also consider an optimal routing for Gravity TMs (the reverse would not work, as routing for bimodal TMs are defined only on subset of OD pairs). That is, a routing $\mathbf{f}$ such that $\text{OPTU}(\mathbf{D}_G) = \text{PERF}(\mathbf{f}, \{\mathbf{D}_G\})$ (where $\mathbf{D}_G$ is the gravity TM). This routing is computed by solving a multi-commodity flow LP with respect to the TM $\mathbf{D}_G$.

Results for a representative sample of topologies and TMs are shown in Figure 1. The figures plot the performance ratio of the different routings as a function of the margin w. For all routing (as should be), the performance ratio (which measures the worst ratio on the set of TMs) increases with the margin w (as the set of TMs expands). Two of the routings, opt and no-margin-opt have an (optimal) performance ratio of 1 when $w = 1$.

We observe that the routing optimized for the set of TMs D_w significantly outperforms the other routings we evaluated. Note that for larger margins (say in the range 4–10), the best possible performance guarantee on the set D_w (that is, the performance ratio of opt) often approaches the optimal oblivious ratio (hence, for this amount of uncertainty one might as well use global-opt instead of opt). The worst performers are the routings that are not even optimized for the base TM $\mathbf{D}$, OSPF and nm-gravity-opt on the Bimodal demands. It is interesting to note that even when these routings happened to perform well on the base TM $\mathbf{D}$, the performance guarantees still deteriorates quickly with the margin w. Another interesting observation is that the no-margin-opt routing which starts out with *optimal performance ratio* of 1 for $w = 1$, quickly degrades with the error margin, in some cases, under-performing OSPF routing for larger margins.[3] This behavior indicates that it is important to take into account error margins when optimizing a routing for specific TMs.

The optimal oblivious routing (global-opt) exhibits different behavior patterns. On about half the topologies/base TMs, it shows close to its global worst-case ratio even on smaller margins. On others, it performs well on small margins, but eventually (with margin 1.5-2.5 obtains its near-worst case ratio).

We also observe that the optimal routing (opt) generally allows for fairly sizable margins (over 50%, that is $w \geq 1.5$) with performance ratio that is close to 1. Figure 2 summarizes the fraction of topologies/base TMs that can tolerate a certain error margin while guaranteeing a certain performance ratio. The two performance ratios considered are 1.05 (guaranteed to be within 5% of optimal maximum utilization) and 1.25 (guaranteed to be within 25% of optimal). The figure shows the two cumulative fraction plots. The optimal routing, in most cases, can have a margin of 50% ($w = 1.5$) with 5% performance overhead (performance ratio at most 1.05) and a margin of 100% ($w = 2$) with 25% overhead (performance ratio of at most 1.25). The no-margin-opt and OSPF routings do not perform nearly as well: With OSPF routing, for the vast majority of instances, the performance ratio exceeds 1.25 with margins smaller than 10% ($w = 1.1$). The no-margin-opt deteriorates quickly and on most instances has a performance ratio that exceeds 1.25 for margins that are at most 30% ($w = 1.3$).

Our observations from the experiments are consistent across the different topologies and TM generation methods. This indicates that our conclusions are not likely to be sensitive to various inaccuracies in our data (that stem from inaccurate maps, heuristic capacities, and heuristic TMs).

We observed that all our test networks have an optimal oblivious ratio smaller than 2. What can we take from this observation ? Can we expect it to prevail when network sizes scale up? It is known that some graphs with *asymmetric* link capacities have optimal oblivious ratio that is $\Omega(n^{0.5})$ [3] (n is the number of nodes in the graph). Räcke's has established [17] that the worst case for "symmetric" capacities is at most $O(\log^3 n)$. It is also known that some families of symmetric graphs have optimal oblivious ratio of $\Omega(\log n)$. Thus, the optimal oblivious ratio of *arbitrary* sym-

[3] A natural question is by how much in the worst case can the performance ratio deteriorate as the margins increase. It is not hard to see that for any $\mathbf{f}$ and $\mathbf{D}$ we have $\text{PERF}(\mathbf{f}, D_w) \leq w^2\text{PERF}(\mathbf{f}, \mathbf{D})$ (and that this is asymptotically tight).

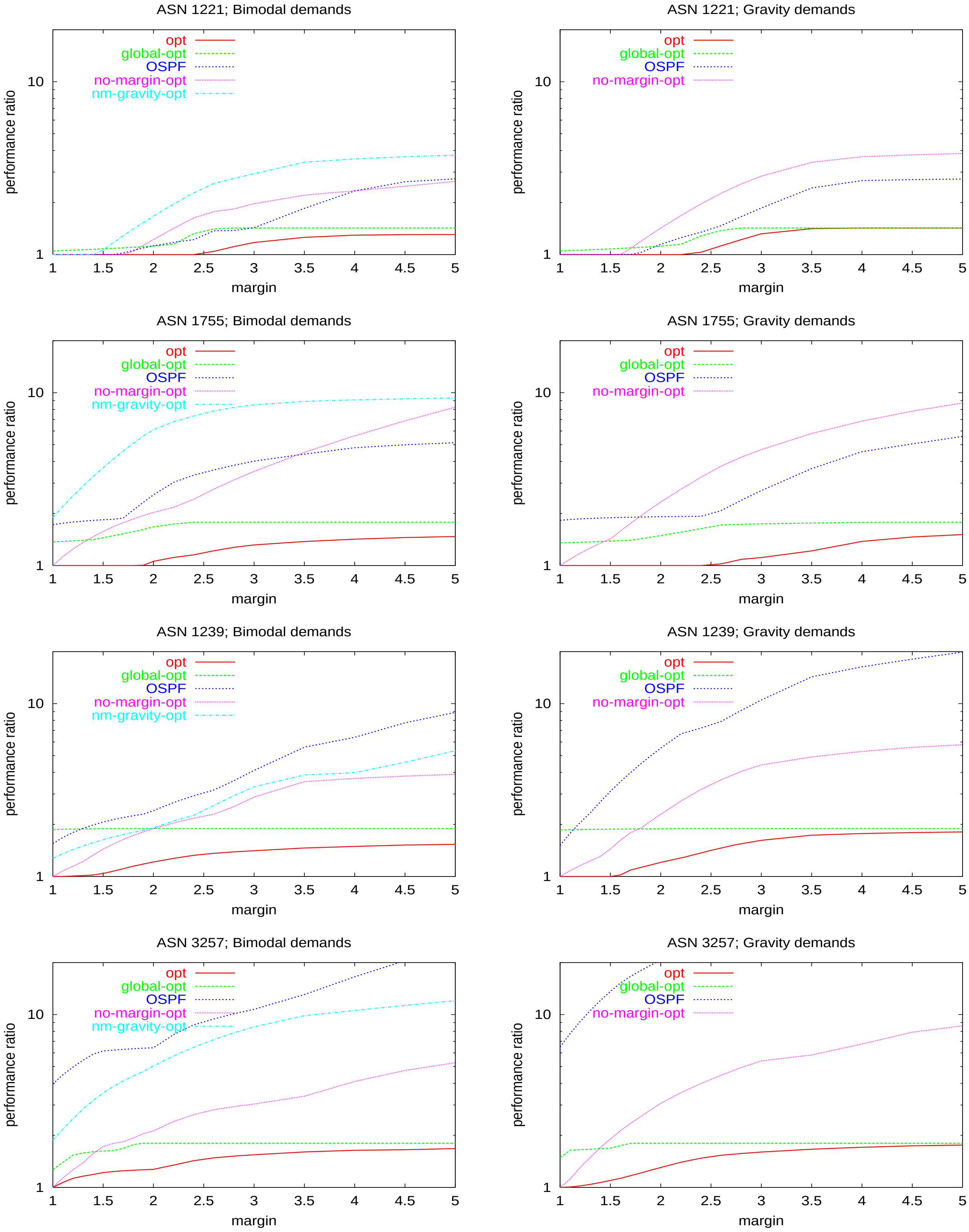


Figure 1: Performance ratio versus margin for several routing strategies, topologies, and TMs.

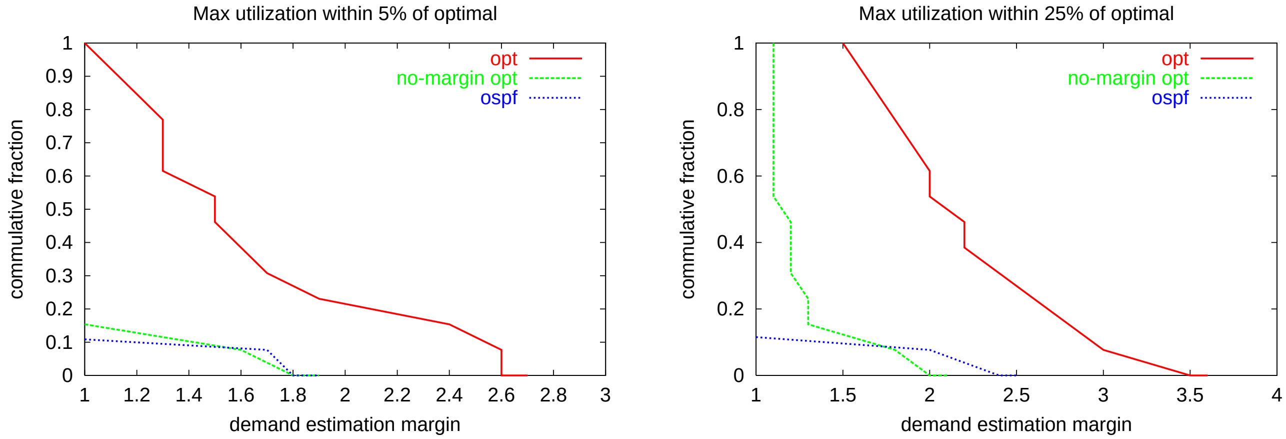


Figure 2: Cumulative fraction of networks/TMs for which a performance ratio of under 1.05% or 1.25% was obtained below a certain error margin.

metric networks can grow logarithmically as they scale up. Some supporting evidence, however, for the "asymptotics" of our observation is that two natural and very different families of graphs: the cliques (all complete graphs on n nodes) and the cycles (all cyclic paths on n nodes), have optimal oblivious ratio of $2 - 2/n$, that is, the optimal oblivious ratio remains smaller than 2 when the network sizes grow (see Section 6).

Performance under link failures. We performed a preliminary evaluation of the performance of different routings under link failures. Since each PoP to PoP link typically corresponds to several physical links, we simulated link failures where a random link "loses" 50% or 80% of its capacity. Table 3 shows the performance ratio of different routings on the "failed" network. The set of TMs considered are the gravity TMs with an error margin of 2.5. The routings considered are (i) the optimal routing on the failed network, (ii) the optimal routing on the original network, where only flow that traverses a failed link is rerouted, and the rerouted flow is routed proportionally to the unaffected part of the routing of the same OD pair, (iii) the OSPF routing. These results indicate that the optimal routing for the non-failed network, although not optimized for failures, typically outperforms OSPF routing under failures. The explanation is that a good "oblivious" routing tends to use many available paths for each OD pair, a property that increases its resilience to failures. An interesting open question is to design and evaluate routings that have guaranteed restoration performance (eg, optimize performance under the constraint that only affected flows are rerouted).

5. LP MODELS

We start by stating some lemmas we used for reducing the size of the LP model. We then summarize a recent result [3] which established that an optimal oblivious routing (and the oblivious ratio) of a network can be computed in polynomial time in the size of the network. We then develop a simplified LP model that achieves considerably faster running times, and adapt this model to handle interval restrictions on OD-pair demands.

ASN	fail	opt	non-fail-opt	ospf
1755	50%	1.000	1.22	1.916
1755	80%	1.000	2.34	1.916
6461	50%	1.302	2.21	6.878
6461	80%	1.303	4.20	6.878
3967	50%	1.157	1.40	4.537
3967	80%	1.157	2.21	4.537

Table 3: Performance ratio of different routings under link failures. Median performance ratio for failure of random link.

5.1 Basic lemmas

The following lemma shows that for the purpose of computing performance ratio, we can "factor out" parts of the network where path diversity is not possible (thus, all routing would perform the same.). We used this lemma to reduce the size of the input topologies.

LEMMA 5.1. *Removal of degree-one nodes does not affect the oblivious ratio of the network. Similarly, it does not affect the optimal performance ratio with respect to any set of TMs.*

Lemma 5.1 is a corollary of the following Lemma:

LEMMA 5.2. *The optimal oblivious ratio of a network can be computed by partitioning the network to 2-edge-connected components and taking the maximum of the oblivious ratio over these components.*

PROOF. If the network G is not 2-edge connected, it can be partitioned to two non-empty components A and B that are connected by an edge (a, b) where $a \in A$ and $b \in B$. It is easy to see that the optimal oblivious ratio of G is at least that of the maximum optimal oblivious ratios of A and B: The optimal performance ratio obtained on G for TMs that have positive demands only at OD pairs that both lie in A (respectively, both lie in B) is equal to the optimal oblivious ratio of A (respectively, B). To see that, observe that all flow leaving/entering A must go through the edge (a, b), thus there is never an advantage to route demand internal to A through the edge (a, b) and out of A, since this flow will have to traverse back on the same edge and form a flow cycle (the symmetric argument holds for B). The optimal oblivious ratio of G is at least the optimal performance ratio on these more restricted set of TMs.

We now argue the converse, that the optimal oblivious ratio on G is at most the maximum optimal oblivious ratio of A and B. Let $\mathbf{f}_A$ (respectively, $\mathbf{f}_B$) be an optimal oblivious routing on A (respectively, B). We extend the routings $\mathbf{f}_A$ and $\mathbf{f}_B$ to a routing $\mathbf{f}_G$ on G as follows: all OD pairs internal to A or B are routed according to the respective routing. The routing for OD pair (a', b') where $a' \in A$ and $b' \in B$ (similar construction for pairs (b', a')) is routed by concatenating the routing $\mathbf{f}_A$ from a' to a with a flow of value 1 from a to b with the routing $\mathbf{f}_B$ from b to b'. Consider now a TM $\mathbf{D}_G$ on G. We will show that the performance ratio of $\mathbf{f}_G$ on G is at most the maximum optimal oblivious ratio of A and B. We can assume (by scaling $\mathbf{D}_G$) that the maximum edge utilization of the optimal routing of $\mathbf{D}_G$ is 1. Thus, the performance ratio of $\mathbf{f}_G$ on $\mathbf{D}_G$ is equal to the maximum edge utilization. We now define the TMs $\mathbf{D}_A$ and $\mathbf{D}_B$ for A and B, respectively; where $\mathbf{D}_A$ is obtained by aggregating all the demand into/from nodes in A from/into nodes in B to demands from/into the node a ($\mathbf{D}_B$ is similarly defined).

The maximum edge utilization of $\mathbf{f}_G$ on $\mathbf{D}_G$ is the maximum utilization over the edges of A, the edges of B, and the edge (a, b). The utilization of the edge (a, b) is equal to the aggregated demand between A and B. Since the utilization must be at least that also for the optimal routing for $\mathbf{D}_G$, from our scaling assumption it follows that the aggregated demand is at most the capacity of (a, b), and thus the utilization is at most 1. The maximum edge utilization over the edges of A is equal to the utilization of $\mathbf{f}_A$ on the demands $\mathbf{D}_A$, which is at most the optimal oblivious ratio of A (similar for B.) The symmetric argument for the edges of B concludes the proof. □

The following Lemma states that the optimal oblivious ratio of a network with symmetric directed links (that is, the link capacities are equal in both direction) is the same as the oblivious ratio of an undirected network derived from it by replacing each set of directed links by a single undirected link with the same capacity. This lemma says that known bounds for undirected graphs carry over to "real" networks (where links are directed and symmetric). We also use this lemma for reducing the size of our LP models.

LEMMA 5.3. *Consider an undirected network G, and a directed network G' derived from it by replacing each edge e by two anti-parallel arcs that have the same capacity as e. The two networks, G and G' have the same optimal oblivious ratio. Moreover, G and G' have (the same) symmetric optimal oblivious routing.*

5.2 The LP model of [3]

It was shown in [3] that an optimal oblivious routing can be computed by solving a Linear Program (LP) with a polynomial number of variables, but infinitely many constraints (for every possible TM there is a set of constraints). We refer to this LP in the sequel as the "master LP." We use the following notation: the term "link" for an undirected edge, "edge" for a directed edge, and let link-of(e) be the link corresponding to edge e. We use the notation

$$f_{ij}(l) = \sum_{e:\text{link-of}(e)=l} f_{ij}(e) \ .$$

We learn from Lemma 5.3 that the routing problem on symmetric directed networks (where the two directions of each link have the same capacity) can be reduced to one on "undirected" networks. We use this equivalence to reduce the number of variables in our LP models.[4]

$$\begin{aligned}
&\min r \\
&f_{ij}(e) \text{ is a routing} \\
&\forall \text{ links } l, \forall \text{ TMs } \mathbf{D} \text{ with } \textsc{optU}(\mathbf{D}) = 1: \\
&\qquad \sum_{ij} f_{ij}(l) d_{ij} / \text{cap}(l) \le r \qquad (1)
\end{aligned}$$

Furthermore, given a routing $f_{ij}(e)$, the constraints (1) can be tested by solving, for each link l, the following "slave LP," and testing if the objective is $\le r$ or not.

$$\begin{aligned}
&\max \sum_{ij} f_{ij}(l) d_{ij} / \text{cap}(l) \qquad (2) \\
&g_{ij}(e) \text{ is a flow of demands } \ d_{ij} \\
&\forall \text{ links } m\text{:} \ \sum_{ij} g_{ij}(m) \le \text{cap}(m) \\
&\forall \text{ demands } i \to j\text{:} \ d_{ij} \ge 0
\end{aligned}$$

Thus, the LPs (2) can be used as a separation oracle for the constraints (1), giving polynomial solvability using the Ellipsoid algorithm [12].

5.3 Deriving a simpler LP model

We derive a simpler LP model that enables us to efficiently process larger networks. For presentation simplicity, our discussion focuses on computing the optimal *oblivious routing*, that is, a routing that provides performance guarantees with respect to all possible TMs. We then state the generalized LP model we used to support interval restrictions on OD pairs demands.

The first simplification one might try to apply is to somehow directly combine the master and slave LPs, to yield a single polynomial size LP instance. However, there are two obstacles: first, both $f_{ij}(l)$ and d_{ij} would be variables in a combination, resulting in quadratic (non linear) constraints, and second, requiring that a maximum over an LP be $\le r$ is not readily modeled. Fortunately, the LP dual of the slave systems (2) leads to a nice characterization:

THEOREM 1. *A routing $f_{ij}(e)$ has oblivious ratio $\le r$ if and only if there exist weights $\pi(l, m)$ for every pair of links l, m such that*

P1 $\sum_m$ *cap*$(m)\pi(l, m) \le r$ *for every link l*

P2 For every link l, for every demand $i \to j$, and for every path $h_1, h_2, \ldots, h_p$ from i to j,

$$f_{ij}(l) \le \mathit{cap}(l) \sum_{k=1}^{p} \pi(l, \mathit{link\text{-}of}(h_k)) \ .$$

P3 $\pi(l, m) \ge 0$ *for all links l, m*

[4] To simplify our presentation we discuss "undirected" networks, but, similarly to the model in [3], our models can be extended to directed-asymmetric networks (with a 2-fold increase in the size of the LPs).

PROOF. The proof is essentially duality applied to the slave problem. Requirements (P1)-(P3) are equivalent to stating that the slave LP's have dual objective values $\le r$.

("if" direction): Let $f_{ij}(e)$ be a routing, and $\pi(l,m)$ be weights satisfying requirements (P1)-(P3). Suppose (g,d) is a flow of demands d with maximum utilization of 1, and let l be a link. For each demand $i \to j$, g_{ij} must contain paths from $i \to j$ of total weight d_{ij}. From (P2) and (P3), summing over all paths, we have

$$f_{ij}(l)d_{ij} \le \text{cap}(l)\sum_{h} \pi(l,\text{link-of}(h))g_{ij}(h) \ .$$

Summing over all demands $i \to j$, we have

$$\begin{aligned}\sum_{ij} f_{ij}(l)d_{ij} &\le \text{cap}(l)\sum_{ij}\sum_{h} \pi(l,\text{link-of}(h))g_{ij}(h)\\ &= \text{cap}(l)\sum_{m}(\pi(l,m)\sum_{ij} g_{ij}(m))\\ &\le \text{cap}(l)\sum_{m}\pi(l,m)\text{cap}(m)\end{aligned}$$

The last inequality follows since g fits within the edge capacities ($\sum_{ij} g_{ij}(m) \le \text{cap}(m)$), and from (P1)

$$\sum_{ij} f_{ij}(l)d_{ij} \le \text{cap}(l)\sum_{m}\pi(l,m)\text{cap}(m) \le \text{cap}(l)r \ .$$

This says that for any demands d which can be routed with congestion 1, f's utilization on any link l is at most r, which is what we wanted.

("only if" direction): Let flow $f_{ij}(e)$ have oblivious ratio $\le r$, and let l be a link. The dual of the slave LP (2) for link l is:

$$\begin{array}{lr}\min \sum_{m} \text{cap}(m)\pi(l,m) & (3)\\ \forall \text{ demands } i \to j\text{: } \lambda_{ij}(l,j) \ge f_{ij}(l)/\text{cap}(l) & \\ \forall \text{ demands } i \to j,\ \forall \text{ edges } e = i' \to j'\text{:} & \\ \quad \pi(l,\text{link-of}(e)) + \lambda_{ij}(l,i') - \lambda_{ij}(l,j') \ge 0 & (4)\\ \forall \text{ links } m\text{: } \pi(l,m) \ge 0 & \\ \forall \text{ demands } i \to j,\ \forall \text{ nodes } k\text{: } \lambda_{ij}(l,k) \ge 0 & \\ \forall \text{ demands } i \to j\text{: } \lambda_{ij}(l,i) = 0 & \end{array}$$

The variable $\lambda_{ij}(l,k)$ is the dual multiplier on the flow conservation constraint for demand $i \to j$ at node k. Since there is no flow conservation constraint in the primal at node i, we have introduced $\lambda_{ij}(l,i)$, fixed at 0, for convenience. The variable $\pi(l,m)$ is the dual multiplier on the capacity constraint for link m.

Since $f_{ij}(e)$ has oblivious ratio $\le r$, the primal slave LP for any link l must have optimum $\le r$, and hence also the dual slave LP for link l must have optimum $\le r$. Hence, the $\pi(l,m)$ from the dual slave LPs satisfy (P1). Trivially, they also satisfy (P3). Now, let $i \to j$ be a demand, and $h_1,\ldots,h_p$ be a path from i to j. Summing up constraint (4) over edges $h_1,\ldots,h_p$, we have

$$\sum_{k=1}^{p} \pi(l,\text{link-of}(h_k)) + \lambda_{ij}(l,i) - \lambda_{ij}(l,j) \ge 0$$

Since $\lambda_{ij}(l,i) = 0$,

$$\sum_{k=1}^{p} \pi(l,\text{link-of}(h_k)) \ge \lambda_{ij}(l,j) \ge f_{ij}(l)/\text{cap}(l)$$

so the $\pi(l,m)$ satisfy (P2). □

We next apply Theorem 1 to show that the problem can be solved by a single polynomial-sized LP. This results in a significant algorithmic performance gain, since it means the problem can be solved by the more efficient Interior-Point algorithm [13].

THEOREM 2. *The oblivious ratio of a network can be computed by a single LP with $O(mn^2)$ variables and $O(nm^2)$ constraints.*

PROOF. We introduce the variables $p_l(i,j)$, for each link l and OD pair i,j. The variable $p_l(i,j)$ is the length of the shortest path from i to j according to the link weights $\pi(\ell,m)$ (for all m). The introduction of these variables allows us to replace the exponential number of constraints (for all possible paths) in Requirement (P2) of Theorem 1 with a small polynomial number of constraints.

$$\begin{array}{lr}\min r & (5)\\ f_{ij}(e) \text{ is a routing} & \\ \forall \text{ links } l\text{: } \sum_{m} \text{cap}(m)\pi(l,m) \le r & \\ \forall \text{ links } l,\ \forall \text{ pairs } i \to j\text{:} & \\ \quad f_{ij}(l)/\text{cap}(l) \le p_l(i,j) & \\ \forall \text{ links } l,\ \forall \text{ nodes } i,\ \forall \text{ edges } e = j \to k\text{:} & \\ \quad \pi(l,\text{link-of}(e)) + p_l(i,j) - p_l(i,k) \ge 0 & \\ \forall \text{ links } l,\ m\text{: } \pi(l,m) \ge 0 & \\ \forall \text{ links } l,\ \forall \text{ nodes } i\text{: } p_l(i,i) = 0 & \\ \forall \text{ links } l, \forall \text{ nodes } i,j\text{: } p_l(i,j) \ge 0 & \end{array}$$

This LP has $O(mn^2)$ variables and $O(nm^2)$ constraints. □

5.4 Interval restrictions on OD demands

To compute the oblivious ratio when demand $i \to j$ is restricted to the range $[a_{ij}, b_{ij}]$, we modify the slave LP (2) by replacing the constraint $d_{ij} \ge 0$ with $a_{ij} \le d_{ij} \le b_{ij}$, and following that change through the dual LP (3) into the single LP (5). This results in the introduction of the slack variables $s_l^-(i,j)$ and $s_l^+(i,j)$ for the lower and upper bound constraints on d_{ij}.

$$\begin{array}{l}\min r\\ f_{ij}(e) \text{ is a routing}\\ \forall \text{ links } l\text{: } \sum_{m} \text{cap}(m)\pi(l,m) \le r\\ \forall \text{ links } l,\ \forall \text{ pairs } i \to j\text{:}\\ \quad f_{ij}(l)/\text{cap}(l) - s_l^+(i,j) + s_l^-(i,j) = p_l(i,j)\\ \forall \text{ links } l,\ \forall \text{ nodes } i,\ \forall \text{ edges } e = j \to k\text{:}\\ \quad \pi(l,\text{link-of}(e)) + p_l(i,j) - p_l(i,k) \ge 0\\ \forall \text{ links } l\text{:}\\ \quad \sum_{ij}(b_{ij}s_l^+(i,j) - a_{ij}s_l^-(i,j)) \le 0\\ \forall \text{ links } l,\ m\text{: } \pi(l,m) \ge 0\\ \forall \text{ links } l,\ \forall \text{ nodes } i\text{: } p_l(i,i) = 0\\ \forall \text{ links } l,\ \forall \text{ nodes } i,j\text{: } p_l(i,j) \ge 0\\ \forall \text{ links } l,\ \forall \text{ nodes } i,j\text{: } s_l^-(i,j) \ge 0\\ \forall \text{ links } l,\ \forall \text{ nodes } i,j\text{: } s_l^+(i,j) \ge 0\end{array}$$

This reduces to the single LP (5) if the bounds are $[0,\infty)$.

6. CLIQUES AND CYCLES

We analyze the optimal oblivious ratio for two simple families of network topologies: The *cycle* topology C_n has n nodes that are connected in a cycle pattern with unit capacity links. The *clique* topology K_n has n nodes connected via a complete graph, that is, there is a unit capacity edge connecting any two nodes.

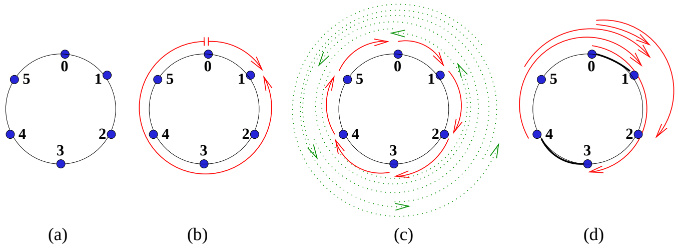
Our interest in these networks is two-fold. First, we shall see that these networks admit an optimal oblivious ratio bounded by 2 (even for large values of n). This is in agreement with the ratio computed for our ISP networks, and provides some indication that a small "constant" optimal oblivious ratio is possible as ISP networks scale up. Second, we also use these simple topologies to better illustrate to the reader our metrics and notion of a good "demand oblivious" routing.

These two families of topologies are highly homomorphic ("look the same" from any node). We will find the following lemma useful for analyzing them:

LEMMA 6.1. *If two nodes u and v are homomorphic under some homomorphism H, then there exists an optimal oblivious routing such that the routing from u to a node w on an edge e is equal to the routing from $v = H(u)$ to $H(w)$ on the edge $H(e)$.*

Figure 3 uses the C_6 topology (Figure 3 (a)) to illustrate the issues in selecting a good routing. Consider first a TM that constitutes of a positive demand on the single OD pair $0 \rightarrow 1$. The optimal routing for this TM (that is, the routing that minimizes the maximum utilization) balances the load on the two disjoint paths from 0 to 1: half the demand flows on the direct edge $(0, 1)$ and the other half on the 5-edge path $(0, 5, 4, 3, 2, 1)$ (this "even-split" routing is illustrated in (b)). Observe that the shortest-path routing, which sends the flow of each OD pair on the shorter of the two available paths (that is, for our TM it would send all flow on the direct edge $(0, 1)$), has performance ratio of 2 on our TM, as the maximum link utilization obtained by this routing is double that of the even-split routing. We next consider another simple TM where there are unit demands on all "consecutive" OD pairs $i \rightarrow (i + 1) \bmod 6$ (for $i = 0, \ldots, 5$), and no demand on other pairs. We first consider routing the demand of each OD pair evenly on the two available paths (using the "even-split" routing we used in part (b)). The flow routes of the routings on this TM are illustrated in part (c) of the figure: The solid lines indicate the routes used by the shortest-path routing. The solid and dotted paths together are used by the even-split routing, which induces a flow of half from every OD pair demand on every edge. We thus obtain that the even-split routing has link utilization of 3. In contrast, the shortest-path routing (solid lines in (c)) would send on each edge only the demand due to the very same OD pair, resulting in maximum link utilization of 1. Thus, the performance ratio of the even-split routing on this TM is at least 3 (in fact, it is exactly 3 since the direct routing is optimal). The same argumentation can be carried over to other cycle topologies C_n; it is not hard to see that the even-split routing would have utilization of $n/2$ and performance ratio of $n/2$ (since the shortest-path routing has maximum utilization of 1).

What this means is that the even-split routing is a bad oblivious routing. The next question to ponder about is finding a good oblivious routing. We now consider general TMs and argue that the shortest-path routing has oblivious ratio of 2. The shortest-path routing for all OD pairs that utilize the edge $(0, 1)$ is illustrated in part (d) of the figure. Consider an arbitrary TM, and the edge with highest utilization according to the shortest-path routing. Without loss of generality we can assume that this edge is $(0, 1)$. We refer to the edge $(3, 4)$ as the "opposite edge" from $(0, 1)$. (In general for even valued n, the opposite edge of $(i, (i + 1) \bmod n)$ on C_n is the edge $(i + n/2, (i + 1 + n/2) \bmod n)$.) It is not hard to verify the following property: for every OD pair that its demand is routed by the shortest-path routing on the edge $(0, 1)$, the two edges $(0, 1)$ and its opposite edge $(3, 4)$ "disconnect" the pair (the two edges are "cut" edges). What this means is that for any routing, the sum of flows that are due to demand $0 \rightarrow 1$ on the edge $(0, 1)$ and its opposite edge must be at least the demand between 0 and 1. It follows that for any routing, the maximum utilization over the two edges $(0, 1)$ and its opposite must be at least half of the utilization of $(0, 1)$ under shortest-path routing. We thus obtain that the performance ratio of the shortest-path routing is at most 2.



(a) (b) (c) (d)

Figure 3: (a) The C_6 topology. (b) The "even-split" routing which is optimal for single OD pair unit demand $0 \rightarrow 1$. (c) For unit demands on all OD pairs $i \rightarrow i+1$, the even-split routing has maximum utilization of $n/2$, whereas the optimal routing has utilization 1. (d) The shortest-path routing for OD pairs using the edge $(0, 1)$.

We next provide a formal proof that states that the best possible performance ratio for the cycle C_n is $2 - 2/n$. We shall see that the optimal oblivious routing will send some flow along the longer path (but most flow along the shorter path). We use the following notation: we number the nodes of C_n as $0, \ldots, n - 1$, with node numbers taken modulo n, so that the edges are $(i, i + 1)$ and $(i + 1, i)$. $f_{a,b}(i, j)$ is the oblivious routing flow for demand $a \rightarrow b$ on edge $i \rightarrow j$.

LEMMA 6.2. *The optimal oblivious performance ratio for C_n (the cycle on n vertices) is $2 - 2/n$.*

PROOF. We first show that the ratio is at least $2 - 2/n$: From symmetry (see Lemma 6.1) and flow conservation considerations, there is an optimal oblivious routing of the following form: for some $x_i \geq 0$,

$$\begin{aligned}
&\forall a,k \in [1,n-1], \forall i \in [0,k-1]: \\
&\qquad f_{a,a+k}(a+i,a+i+1) &=& \quad x_k \\
&\forall a,k \in [1,n-1], \forall i \in [0,n-k-1]: \\
&\qquad f_{a,a+k}(a-i,a-i-1) &=& \quad 1-x_k \\
&\qquad \text{for all other } a,b,c,d: \\
&\qquad f_{a,b}(c,d) &=& \quad 0 \\
&\qquad \forall k \in [1,n-1]: \\
&\qquad x_k &=& \quad 1-x_{n-k}
\end{aligned}$$

For any a, a demand $a \to (a+1)$ of size 2 can be routed within unit capacities, so from the load on edge $(a,a+1)$, we have that the optimal ratio is at least

$$2 * f_{a,a+1}(a,a+1) = 2x_1 \ .$$

On the other hand, a demand for all $a \to a+1$ ($a = 0 \ldots n-1$) of size 1 can also be routed within unit capacities, so from the load on a particular edge $(a,a+1)$, we have that the ratio is at least

$$\sum_{i=0\ldots n-1} f_{a-i,a-i+1}(a,a+1) = x_1 + (n-1)(1-x_1) \ .$$

From the above two bounds we obtain that the optimal ratio is at least

$$\max\{2x_1, x_1 + (n-1)(1-x_1)\} \geq 2 - 2/n \ .$$

(the maximum is minimized when $x_1 = (n-1)/n$).

It remains to show that the optimal ratio is at most $2 - 2/n$. Consider the routing obtained by setting $x_k = (n-k)/n$, we show that this routing has oblivious performance ratio of at most $2-2/n$. Consider, without loss of generality, the edge $(0,n-1)$. A demand $d_{a,b}$, with $0 \leq a < b \leq n-1$, must either be routed using the edge $(0,n-1)$, or be routed on the path $(a,a+1,\ldots,b)$, using $b-a$ edges. Similarly, a demand $d_{b,a}$ with $0 \leq a < b \leq n-1$ must either be routed using the edge $(n-1,0)$ or be routed on the path $(b,b-1,\ldots,a)$ using $b-a$ edges. Consider now a TM that can be routed such that each edge has at most 1 unit of flow on it. It suffices to show that our routing has utilization at most $1-2/n$ on that TM. Consider such demands and supposed that the optimal routing for that TM is such that none of the demands were routed on the edge $(0,n-1)$. Then the total edge load generated on the edges $(0,1,\ldots,n-1)$ would be:

$$\sum_{0\leq a<b\leq n-1} (b-a)(d_{a,b}+d_{b,a}) \ .$$

However, at most a total flow of 1 can be routed using the edge $(n-1,0)$, so the combined flow on the other $n-1$ edges must be at least

$$(\sum_{0\leq a<b\leq n-1} (b-a)(d_{a,b}+d_{b,a})) - (n-1)$$

The total flow on those $n-1$ edges must be less than their total capacity, so we obtain that

$$(\sum_{0\leq a<b\leq n-1} (b-a)(d_{a,b}+d_{b,a})) - (n-1) \leq n-1 \ .$$

The utilization on edge $(0,n-1)$ of our oblivious routing for these demands is

$$\begin{aligned}
&\sum_{0\leq a<b\leq n-1} (1-x_{b-a})(d_{a,b}+d_{b,a}) \\
&= \sum_{0\leq a<b\leq n-1} (b-a)/n(d_{a,b}+d_{b,a}) \\
&\leq \ 2(n-1)/n = 2-2/n \ .
\end{aligned}$$

□

For the clique topology, the shortest-path routing, where the flow of each demand is routed on the direct edge, performs very poorly, with performance ratio of $n-1$. We shall see that the optimal oblivious routing for the clique topology utilizes 2-hop paths.

Lemma 6.3. *The optimum oblivious ratio for K_n (the complete graph on n vertices) is $2-2/n$.*

Proof. We first show that the ratio is at least $2-2/n$: From symmetry (see Lemma 6.1) and flow conservation, we know that there is an optimal oblivious routing with the following form: for some $x \geq 0$,

$$\begin{aligned}
\text{for all distinct } a,b\text{: } f_{a,b}(a,b) &= x \qquad (6)\\
\text{for all distinct } a,b,c\text{: } f_{a,b}(a,c) &= \\
f_{a,b}(c,b) &= (1-x)/(n-2) \\
\text{for all other } a,b,c,d\text{: } f_{a,b}(c,d) &= 0
\end{aligned}$$

The minimum s-t cut between any two nodes is $n-1$. Thus, for any given OD pair a,b, a demand $a \to b$ of size $(n-1)$ can be routed such that the maximum flow on any edge is 1. By considering such single OD pair demands $a \to b$, and the edge (a,b), we obtain that the optimal ratio is at least

$$(n-1) * f_{a,b}(a,b) = (n-1) * x \ .$$

We now consider a TM such that there is a demand of 1 for each OD pair $c \to d$ ($c < d$). Such TM can also be routed within unit capacities by routing each demand $c \to d$ on the "direct" edge (c,d). By considering the flow of our routing on the edge $a \to b$, we have that the optimal ratio is at least

$$\begin{aligned}
1 * f_{a,b}(a,b) + \sum_{c\notin\{a,b\}} f_{a,c}(a,b) + \sum_{c\notin\{a,b\}} f_{c,b}(a,b) \quad &= \\
x + 2*(n-2)*(1-x)/(n-2) \ .&
\end{aligned}$$

from the above two constraints we obtain that the optimal ratio is at least $\max\{x*(n-1), 2-x\} \geq 2-2/n$ (the maximum is minimized when $x = 2/n$).

It remains to show that the optimal ratio is at most $2-2/n$. We will use a routing of the form (6) with $x = 2/n$, and show that its oblivious performance ratio is at most $2-2/n$. Consider a particular edge $a \to b$. Since a and b have degree $n-1$, any TM which can be routed with at most one unit of flow on each edge must satisfy

$$\begin{aligned}
\sum_{c\neq a} d_{a,c} + d_{c,a} &\leq \quad n-1 \\
\sum_{c\neq b} d_{b,c} + d_{c,b} &\leq \quad n-1
\end{aligned}$$

Therefore,

$$2d_{a,b} + 2d_{b,a} + \sum_{c\notin\{a,b\}} (d_{a,c}+d_{c,a}+d_{b,c}+d_{c,b}) \leq 2(n-1)$$

From (6), some optimal oblivious routing then satisfies that the flow on the edge (a, b) is equal to

$$\sum_{c,d} (d_{c,d} f_{c,d}(a, b) + d_{c,d} f_{c,d}(b, a))$$

$$= \frac{\sum_{c \neq a,b} (d_{a,c}(1-x) + d_{b,c}(1-x) + d_{c,a}(1-x) + d_{c,b}(1-x))}{n-2}$$

$$+ d_{a,b} x + d_{b,a} x$$

By substituting $x = 2/n$ we obtain that the flow on (a, b) is

$$\frac{\sum_{c \neq a,b} (d_{a,c} + d_{b,c} + d_{c,a} + d_{c,b}) + (2d_{a,b} + 2d_{b,a})}{n}$$

$$\leq 2(n-1)/n = 2 - 2/n$$

□

7. CONCLUSION

Traffic demands on IP networks are hard to estimate and are dynamic in nature. Good system engineering thus desires a routing that performs well "independently" of traffic demands (or for a wide range of demands). The goal of our study was to understand the viability of obtaining such a routing, by exploring the tradeoffs between accuracy of TM estimation and attainable utilization performance of the routing. We arrive at perhaps unexpected conclusions.

First, it is possible to obtain a surprisingly good routings with poor or no knowledge of the traffic demands: On current ISP topologies, there exists a routing that guarantees performance ratio that is less than 2 on any possible traffic matrix. This "demand oblivious" routing is designed with no knowledge of the traffic matrix taking only the topology (along with link capacities) into account. With a very limited knowledge of the TM we can do much better, often obtaining a routing with performance ratio that is very close to 1 even for error margins of 50%-100% in knowledge of the traffic demands. Similarly, one can obtain a fixed routing that would perform well on an expected range of demands, thus, reducing the need for routing adjustments when traffic demands shift.

Second, it is unlikely that such a "robust" routing can be obtained via standard previously-existing tools, it seems that obtaining close to optimal performance guarantees *with respect to a range of possible demands* requires the algorithmic tools we developed and employed here: The OSPF routings based on the OSPF weights derived for our test networks performed badly as the set of demands grows. Moreover, and surprisingly so, even a routing designed to be optimal on a specific TM deteriorates quickly with the margins within which the actual demands deviate from the presumed ones.

8. REFERENCES

[1] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao. RFC 2702: Requirements for Traffic Engineering over MPLS, September 1999.

[2] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao. RFC 3272: Overview and Principles of Internet Traffic Engineering, May 2002.

[3] Y. Azar, E. Cohen, A. Fiat, H. Kaplan, and H. Räcke. Optimal oblivious routing in polynomial time. In *Proceedings of the 35th ACM Symposium on the Theory of Computing*, 2003.

[4] S. Bhattacharya, C. Diot, J. Jetcheva, and N. Taft. Geographical and temporal characteristics of inter-POP flows: view from a single POP. In *European transactions on telecommunications*, 2002.

[5] J. Cao, D. Davis, S. V. Wiel, and B. Yu. Time-varying network tomography: router link data. *J. Amer. Statist. Assoc.*, 95:1063–1075, 2000.

[6] Cisco. Configuring OSPF, 1997. `http://www.cisco.com/uni-verc/cc/td/doc/product/software/ios113ed/113ed_cr/np1_c/1cospf.htm`.

[7] CPLEX large-scale mathematical programming software, 2003. `http://www.cplex.com`.

[8] N. G. Duffield and M. Grossglauser. Trajectory sampling for direct traffic observation. *IEEE/ACM Transactions on Networking*, 9:280–292, 2001.

[9] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, J. Rexford, and F. True. Deriving traffic demands for operational IP networks: methodology and experience. *IEEE/ACM Transactions on Networking*, 9:265–279, 2001.

[10] B. Fortz and M. Thorup. Internet traffic engineering by optimizing OSPF weights. In *Proceedings of INFOCOM*, pages 519–528. IEEE, 2000.

[11] B. Fortz and M. Thorup. Optimizing OSPF/IS-IS weights in a changing world. *IEEE journal on selected areas in communications*, 20(4), 2002.

[12] B. Grötschel, L. Lovasz, and A. Schrijver. *Geometric algorithms and combinatorial optimization.* Springer-Verlag, New York, 1988.

[13] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4:373–395, 1984.

[14] R. Mahajan, N. Spring, D. Wetherall, and T. Anderson. Inferring link weights using end-to-end measurements. In *Proceedings of the 2nd Internet Measurement Workshop.* ACM, 2002.

[15] A. Medina, N. Taft, K. Salamatian, S. Bhattacharyya, and C. Diot. Traffic matrix estimation: Existing techniques and new directions. In *Proceedings of the ACM SIGCOMM'02 Conference.* ACM, 2002.

[16] D. Mitra and K. G. Ramakrishna. A case study of multiservice, multipriority traffic engineering design for data networks. In *Proceedings of IEEE GLOBECOM*, pages 1077–1083. IEEE, 1999.

[17] H. Räcke. Minimizing congestion in general networks. In *FOCS 43*, 2002.

[18] E. C. Rosen, A. Viswanathan, and R. Callon. RFC 3031: Multi Protocol Label Switching Architectures, 2001.

[19] M. Roughan, A. Greenberg, C. Kalmanek, M. Rumsewicz, J. Yates, and Y. Zhang. Experience in measuring backbone traffic variability: models, metrics, measurements, and meaning. In *Proceedings of the 2nd Internet Measurement Workshop.* ACM, 2002.

[20] N. Spring, R. Mahajan, and D. Wetherall. Measuring ISP topologies with Rocketfuel. In *Proceedings of the ACM SIGCOMM'02 Conference.* ACM, 2002.

[21] Internet Traffic Engineering Working Group, 2003. `http://www.ietf.org/html.charters/tewg-charter.html`.