Network Sensitivity to Hot-Potato Disruptions

Renata Teixeira
UC San Diego
La Jolla, CA
teixeira@cs.ucsd.edu

Aman Shaikh
AT&T Labs–Research
Florham Park, NJ
ashaikh@research.att.com

Tim Griffin
Intel Research
Cambridge, UK
tim.griffin@intel.com

Geoffrey M. Voelker
UC San Diego
La Jolla, CA
voelker@cs.ucsd.edu

ABSTRACT

Hot-potato routing is a mechanism employed when there are multiple (equally good) interdomain routes available for a given destination. In this scenario, the Border Gateway Protocol (BGP) selects the interdomain route associated with the closest egress point based upon intradomain path costs. Consequently, intradomain routing changes can impact interdomain routing and cause abrupt swings of external routes, which we call *hot-potato disruptions*. Recent work has shown that hot-potato disruptions can have a substantial impact on large ISP backbones and thereby jeopardize the network robustness. As a result, there is a need for guidelines and tools to assist in the design of networks that minimize hot-potato disruptions. However, developing these tools is challenging due to the complex and subtle nature of the interactions between exterior and interior routing. In this paper, we address these challenges using an analytic model of hot-potato routing that incorporates metrics to evaluate network sensitivity to hot-potato disruptions. We then present a methodology for computing these metrics using measurements of real ISP networks. We demonstrate the utility of our model by analyzing the sensitivity of a large AS in a tier 1 ISP network.

Categories and Subject Descriptors

C.2.2 [**Network Protocols**]: Routing Protocols; C.2.3 [**Computer-Communication Networks**]: Network Operations

General Terms

Design, Reliability, Management, Performance

Keywords

Network robustness, sensitivity analysis, hot-potato routing, BGP, IGP, OSPF

1. INTRODUCTION

Internet Service Providers (ISPs) seek to build *robust* networks so that small perturbations in the environment or internal conditions do not significantly impact network performance. However, in practice robustness is not so easily achieved. In fact, robustness is not even an easy thing to measure. A network can seem to be running smoothly only to crash unexpectedly after some seemingly "small" event.

A robust network should have low sensitivity to routing behavior and traffic load variations. To understand the robustness of an IP network, it is important to understand the robustness of both the *control* and *data* planes. For example, a small routing change inside a network may have significant impact in the control plane by causing an extremely large number of routing messages — which in turn may overload and crash routers, even though the routes initially in play were carrying no traffic. On the other hand, small routing shifts of popular routes can impact the data plane by causing a large swing in traffic, perhaps leading to congestion, losses, delay, and jitter.

Even though understanding a network's sensitivity to a change is a crucial step in designing of a robust network, current traffic engineering techniques and tools for network design and management have limited capability to analyze this sensitivity. In fact, there is even little terminology to describe and measure network sensitivity. In this paper, we address this problem by defining metrics for characterizing network sensitivity to routing changes inside an autonomous system's (AS) network.

Routing in an AS is performed by the interplay of interdomain routing, which in the Internet today is BGP [1], and the intradomain routing protocol (such as OSPF [2] and IS-IS [3]). Routers on the border of an AS exchange reachability information using BGP, which is responsible for determining the AS-level forwarding path. Inside each AS, interior gateway protocols (IGPs) determine shortest paths between every router in the network. BGP determines the set of best egress points for a destination prefix. When there are multiple equally good egress points in terms of BGP-specific attributes, IGP provides a ranking of these egress points for a given ingress point in terms of closeness. When the egress point selection is decided by comparing IGP costs, it is commonly called *hot-potato routing*.

In previous work [4], we have used measurements of one ISP network to show that hot-potato routing can have a significant impact on BGP route selection. For example, one IGP routing change was measured to change 62% of a router's BGP table. We call such abrupt BGP route shifts caused by IGP cost changes *hot-potato disruptions*. Motivated by these findings, in this paper we propose metrics that capture the impact of intradomain changes on the control and data planes.

We model the control plane as a *routing matrix* RM that maps a node and a destination to a set of potential egress points[1]. This ma-

SIGCOMM'04, Aug. 30–Sept. 3, 2004, Portland, Oregon, USA.

[1]A node has a set of potential egress points for each destination, because we do not consider the tie-break after the comparison of

trix is a function of the routes sent to an AS and the locally implemented routing policies. The routing matrix stores the set of egress points that are best according to BGP *and* that have equal IGP distances from the node. Changes in both IGP and BGP may affect a network routing matrix. We express this dependence informally in the following relation, where $\bigoplus$ represents the combination of information from the two protocols:

$$RM \leftarrow IGP \bigoplus BGP$$

The data plane is often modeled as a traffic matrix. The *traffic matrix* TM represents the volume of traffic from an ingress point to an egress point. We adopt this abstraction because, when combined with the routing matrix, it is easier to determine metrics such as the overall network load or individual link loads.

We define the traffic demand as a matrix $\mathcal{M}$, which determines the volume of traffic entering the network at a given ingress point to a given destination prefix. The actual egress point that the traffic uses to exit the network is determined by the routing matrix. Thus, changes in traffic demands or routing decisions may result in changes in the traffic matrix. We represent this combination as $\bigotimes$ in the following relation.

$$TM \leftarrow RM \bigotimes \mathcal{M}$$

Informally, we can now talk about measuring robustness, or put another way, the sensitivity of a network (or elements of a network) to perturbations. We are interested in the *control plane sensitivity* to an IGP change ΔIGP. We can capture this as

$$\Delta RM \leftarrow \Delta IGP \bigoplus BGP$$

and say that the network is robust if for all "small" ΔIGP we produce a "small" ΔRM. This naturally extends to a quantification of *data plane sensitivity*, which captures perturbations to the traffic matrix

$$\Delta TM \leftarrow \Delta RM \bigotimes \mathcal{M}$$

and can be derived from the control plane perturbations.

Note that control plane and data plane sensitivities are not necessarily related. For example, a small routing change can produce a large shift in traffic, and a large routing change can leave traffic unchanged. There is an interesting analogy to seismic scales in these definitions. The Richter Scale is a familiar quantification of the magnitude of seismic events. A less familiar scale, called the Modified Mercalli Scale, is used to rank the intensity of a seismic event — that is, the impact that an earthquake has on human society. The magnitude scale captures an intrinsic property of a seismic event, while values on the intensity scale vary with location and may depend on local conditions such as the structural integrity of buildings. So our "control plane sensitivity scale" is analogous to the Richter scale, while the "data plane sensitivity scale" is analogous to the Modified Mercalli Scale.

In this paper, we make the following contributions to the understanding of network sensitivity to internal events:

1. An analytic model of an IP network control plane and data plane, and their interactions;
2. A terminology for network sensitivity analysis;
3. A methodology for computing a network's sensitivity using data typically collected by large ISPs for management purposes; and
4. A case study of applying our model to a large AS of a tier 1 ISP network.

The rest of the paper is organized as follows. We discuss related work in Section 2. In Section 3, we present background material on routing from an ISP perspective and discuss the challenges of modeling it. In Section 4, we model the routing matrix while focusing on a single destination prefix. We then extend the model in Section 5 to capture control plane and data plane sensitivity. Section 6 presents our methodology to compute control plane sensitivity in real networks and analyzes control plane sensitivity of a tier 1 AS to link and router failures. We conclude and present future directions in Section 7.

2. RELATED WORK

Traffic engineering tools evaluate the impact of different network configurations on the traffic matrix. For example, Netscope [5] is a tool from AT&T Labs that allows network operators to experiment with different IGP configurations to determine the load distribution across links. Although Netscope incorporates models of intradomain routing and hot-potato routing, these models are not formally described. The algorithm presented in [6] searches for the set of OSPF weights that leads to an optimal link load distribution. Subsequent work [7] considers weight setting that are more robust to link failures and changes in traffic demands. More recent work [8] models the BGP routing decision in detail and allows the study of changes to BGP configuration on the egress point selection. None of these tools focuses on the network sensitivity to routing changes specifically. They evaluate route selection (RM) and traffic distribution (TM) considering different network settings, but not the impact of the change (ΔIGP or ΔBGP) on the network (ΔTM or ΔRM).

The impact of intradomain routing changes (ΔIGP) on interdomain routing (ΔRM) and that of egress point changes (ΔRM) on traffic (ΔTM) have been quantified using measurements of ISP networks. A study of the Sprint network [9] characterizes the impact of BGP changes on the traffic matrix. This study found that changes in the traffic matrix do not correlate with BGP routing changes. In [4], we measured the impact of intradomain changes on BGP route selection and find that some intradomain changes cause a large churn of BGP routes. This study identified the seriousness of the impact of routing interaction on ISP networks and motivated us to develop the model and analysis presented in this paper.

3. MODELING ROUTING BEHAVIOR

We start this section with an overview of routing in a typical Internet Service Provider (ISP) network. Then, we outline the challenges of modeling the routing behavior and the impact of internal routing changes on the selection of external routes and consequently on traffic.

3.1 Routing within an ISP Network

The Internet is an interconnection of Autonomous System's (AS) networks, which are administered by ISPs and their customers. In order for customers of one ISP to be able to communicate with customers of another, every router in the ISP has to learn how to reach destination prefixes that belong to its customers and those of other ISPs. Typically, BGP is used for exchanging reachability information of external destination prefixes. In particular, routers at the periphery of the ISP speak external BGP (eBGP) with routers outside (from customers or other ISPs) to learn about destination prefixes reachable via these outside routers. If a router at the periphery selects one of the routes learned externally as the best to reach the

IGP distances.

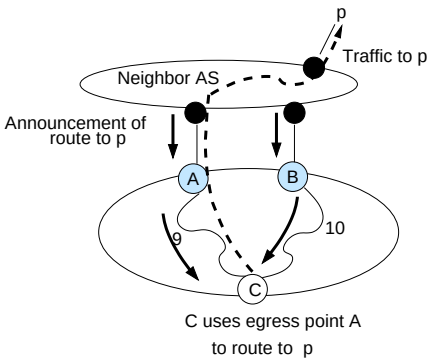

Figure 1: Hot-potato routing determines that C uses route through A to forward packets to the destination prefix p.

destination prefix, it distributes this route to all other routers inside the network using internal BGP (iBGP).

In addition to running BGP (iBGP and eBGP), an AS also runs an Interior Gateway Protocol (IGP) that is responsible for determining the path between routers inside the network. The most commonly used IGPs are OSPF [2] and IS-IS [3], which are link-state protocols. With link-state protocols, each router learns the entire topology of the network and uses Dijkstra's shortest path algorithm to compute the shortest path to all other routers in the network.

An ISP often connects to a customer or another ISP at multiple physical locations. This leads to scenarios where a BGP route for the same prefix is learned by more than one router at the periphery of the ISP network. For example, consider the scenario shown in Figure 1, where routers A and B both learn how to reach prefix p via eBGP. Both routers propagate this route via iBGP to router C (as indicated by the solid arrows). From C's point-of-view, both routes are equally good in terms of their BGP attributes. Therefore, C can choose either A or B as the egress point out of the ISP for sending the traffic destined to p. To break the tie, C selects the router that is *closer* in terms of the IGP distance to the egress point. In this example, C picks A as the egress point for all the traffic to destination prefix p (as shown by the dotted line in Figure 1).

When the BGP best route selection is based on the IGP distance, it is called *hot-potato routing*. In fact, hot-potato routing happens because the IGP distance to the BGP "next-hop" (the egress point) is one of the steps in the decision process used by a router like C to select the best BGP route for a given prefix. Table 1 presents the steps in the BGP decision process. A router uses the decision process on a per prefix basis, i.e., it applies the eight steps outlined in the table for every prefix to select a route out of multiple BGP routes the router learns for that prefix. As the table shows, first the router uses the IGP information to ensure that it can reach the egress point of a route using an IGP path within the network. Then it considers various BGP attributes of routes. At the end of step 5, if two or more routes survive as *equally good* routes, then the router compares the IGP distance to the egress points of these routes, and selects the route with the lowest IGP distance to the egress point.

In transit AS in the core of the Internet a significant fraction of the routes are selected based on hot-potato routing. These include all routes learned from peer networks and from customers that connect to the AS in multiple locations. Hot-potato routing can cause a router to change its routing decision because the IGP distances to egress points have changed due to a failure, a planned maintenance or a traffic engineering event inside the network. We call such BGP routing changes due to the hot-potato rule in the decision process *hot-potato changes*.

0. Ignore if exit point unreachable
1. Highest local preference
2. Lowest AS path length
3. Lowest origin type
4. Lowest MED (with same next-hop AS)
5. eBGP-learned over iBGP-learned
6. Lowest IGP path cost to exit point ("Hot potato")
7. Configuration-specific tie-breaking
(e.g., older route or lowest router-id of BGP speaker)

Table 1: Principle steps in the BGP decision process

Consider Figure 1 again and assume that a failure in one of the links in the path from C to A causes its cost to become 11. This small change in the IGP distance makes egress point B closer to C than A, prompting it to send traffic destined to p via B. Indeed, as observed in [4], in some cases hot-potato routing can cause a large fraction of BGP routes at a given router to switch egress points due to a single IGP change inside the network. Thus, abrupt hot-potato routing changes can have a significant impact on both the control plane and the data plane. The goal of this paper is to understand *when* and *why* such changes happen through an analytic model of the BGP/IGP interaction and sensitivity metrics that characterize network sensitivity to IGP changes.

3.2 Challenges

Modeling the sensitivity of routing and traffic to IGP changes inside an ISP network requires a good understanding of how the network uses IGP and BGP for routing, the resulting routing matrix, the traffic demands, and how the traffic demands and routing matrix combine to generate the traffic matrix. Below we outline the main challenges of modeling this complex system:

IGP. Each AS has the freedom to design its internal routing infrastructure and select the appropriate IGP to route packets within the network. For scalability reasons, both OSPF and IS-IS allow hierarchical routing by dividing a network into areas [2, 3]. When areas are employed, packets are forwarded along shortest paths within an area, but may deviate from the shortest paths when they traverse multiple areas. As a result, a model of hot-potato routing needs to incorporate the link weights and the division of routers into areas and accurately model the IGP path cost computation.

BGP. Modeling the complete set of egress points per prefix within an ISP can become complicated due to the iBGP architecture of a network. The BGP standard requires a full mesh of iBGP connections between all BGP speaking routers in the network [1]. This ensures that all the routers know about the best routes learned by every other router via eBGP. To overcome the resulting scalability issues, networks with a large number of BGP speakers usually structure them into some form of iBGP hierarchy; "router-reflectors" and "confederations" are two popular ways of forming this hierarchy [10]. An unfortunate side-effect is that some BGP routes are available to only a subset of BGP routers inside a network. This can lead to scenarios where different routers in a network have different sets of egress points.

Routing matrix. After we have an abstract model of both IGP and BGP decisions, we still need to combine them to form the routing matrix, which depends on the interaction of the two protocols. Some parts of this interaction are standardized such as steps 0 through 6 of the BGP decision process, whereas some like the tie-breaking rule after the comparison of IGP costs are not. This implies that the interaction between IGPs and BGP depends on router implementation or specific network configuration.

Traffic demands and traffic matrix. The traffic demand placed on a network depends on the traffic sent and received by the end-

users attached to customer and peer networks, and the locations in which they connect to the network. Consequently, traffic demands vary considerably over time. In general, it is hard to obtain a representative snapshot of the traffic demands and the resulting traffic matrix. Even though measurement and estimation of the traffic matrix have received considerable research attention of late [11, 12, 13], measurements are aggregated or sampled and not all routers in the network are capable of collecting the required data [13].

3.3 Assumptions of the Model

To reduce the complexity of modeling all these details, we make the following simplifying assumptions:

1. **All routers know all routes to reach a destination prefix**. As mentioned earlier, this assumption is valid when BGP speakers are organized in a full iBGP mesh. In ASes that use an iBGP hierarchy this assumption no longer holds. However, even in the case of an iBGP hierarchy, the following condition is enough for our model to capture the network sensitivity correctly: every router at the very least learns the route that it would have picked as the best route had it learned all the eBGP-learned routes for a given prefix.

2. **Well-configured iBGP**. Griffin and Wilfong [14] describe scenarios in which iBGP misconfigurations can lead to route deflections. This means that routers in the forwarding path to an egress point for a prefix disagree on the selection of the egress point leading to deflections and loops in an extreme case. We assume that the network configuration satisfies the conditions specified in [14] for avoiding such loops and deflections.

3. **Stable BGP routing snapshots**. We assume that BGP policies and eBGP routes are stable. Our model works with snapshots of BGP routes and analyzes the impact of intradomain routing changes on snapshot of routes.

4. **Stable snapshots of traffic demands**. Similarly, our model takes as input snapshots of the traffic demands to analyze the impact of intradomain routing changes on traffic.

These assumptions allow us to model the state of routing system and snapshots of traffic. Our model does not capture the routing dynamics or traffic load variations directly. A single underlying event may cause a series of routing messages, and routers in the network may take some time to converge to a consistent state. We do not model the sequence of messages exchanged to reach a stable routing state. Instead, we model a routing change from one stable state to another stable state reached after the convergence phase.

4. REGIONS AND REGION SHIFTS

This section first introduces concepts from multidimensional data analysis that inspired us in our network sensitivity analysis. Then, it describes a graph model and the concepts of graph regions and region shifts that together form the basis of our network model of hot-potato routing. The set of definitions presented here capture the interaction between BGP and IGP when considering a single destination prefix. In Section 5, we build on this terminology to define control plane and data plane sensitivity metrics. Table 2 summarizes the notation introduced in this section and the following one, and used in the remainder of the paper.

Basic	
Universe of vertices	$\mathcal{U}$
Undirected weight graph	$G = (V, E, w, d), V \subset \mathcal{U}$
Root set or egress set	$L \subset \mathcal{U}$
v's rank for root vertex l	$d(v, l)$
Region of a root vertex l	$R_l(G, L) \subseteq V$
Region index set of a vertex v	$RI(G, L, v) \subseteq L$
Class of graph transformations	ΔG
Probability function for graph transformations	$\mathbb{P}$
Regional	
Region-shift function	$\mathcal{H}(G, L, v, \delta)$
Vertex sensitivity	$\eta(G, L, \Delta G, \mathbb{P}, v)$
Impact of a graph transformation	$\theta(G, L, \delta)$
Graph sensitivity	$\sigma(G, L, \Delta G, \mathbb{P})$
Control Plane Sensitivity	
Set of destination prefixes	P
Mapping of prefixes to egress sets	$\beta : P \rightarrow 2^{\mathcal{U}}$
Routing-shift function	$\mathcal{H}^{RM}(G, P, v, \delta)$
Node routing sensitivity	$\eta^{RM}(G, P, \Delta G, \mathbb{P}, v)$
Routing impact of a graph transformation	$\theta^{RM}(G, P, \delta)$
Control plane sensitivity	$\sigma^{RM}(G, P, \Delta G, \mathbb{P})$
Data Plane Sensitivity	
Set of ingress nodes	$\mathcal{I}$
Ingress-to-prefix traffic matrix	$\mathcal{M}$
Total traffic volume entering v	$\mathbb{T}(v)$
Traffic-shift function	$\mathcal{H}^{TM}(G, P, \mathcal{M}, v, \delta)$
Ingress node traffic sensitivity	$\eta^{TM}(G, P, \Delta G, \mathbb{P}, \mathcal{M}, v)$
Traffic impact of a graph transformation	$\theta^{TM}(G, P, \mathcal{M}, \delta)$
Data plane sensitivity	$\sigma^{TM}(G, P, \Delta G, \mathbb{P}, \mathcal{M})$

Table 2: Summary of definitions and metrics.

4.1 Multidimensional Data Analysis

Extracting useful information from a network configuration is a very difficult task. We are interested in the sensitivity of a network to hot-potato disruptions, which depends on a number of factors including network topology, routing policies, location in the network, the specific network changes considered, etc. To make matters worse, there seems to be no "one size fits all" metric that can capture the routing sensitivity in a meaningful way. Our approach to managing this complexity is inspired by the area of databases where similar challenges arise — *Online Analytic Processing* (OLAP) [15, 16, 17]. The key to OLAP design is to arrange data in a multidimensional cube, and then provide natural ways to "slice and dice" the data along multiple dimensions.

For instance, imagine a three-dimensional OLAP data cube in which each cell contains sales totals indexed by product sold, city of sales, and day of sales. It is possible to explore the sales data by dicing it into one, two, or three-dimensional slices, and then aggregating and displaying a slice in various ways. For example, if we fix a product and city, then we obtain a one-dimensional slice of the cube that captures the sales over time of the chosen product in the chosen city. This slice could be used to compute an average over all days, or to find the days of maximum sales. In a similar way, if we fix a city, we can obtain a two-dimensional slice of the cube that captures the sales of all products in that city over all days. An

interesting characteristic of most data cubes is that the dimensional data has some natural hierarchical structure. For example, we could generate a new data cube by "rolling-up" the time dimension to the level of *month*, the location dimension to the level of *country*, and the product dimension to the level of *category*.

Figure 2 presents the OLAP data cube we define for the purpose of exploring the impact of hot-potato routing on a network. Our dimensions are *location* (routers), *network change* (representing a fixed class of changes such as "all single link failures" or "all single node failures"), and IP prefixes. We find that a large number of interesting hot-potato sensitivity queries can be computed with a *very simple data cube* — each data cell contains only a single bit! This bit is set if the associated router changes egress points for the associated prefix when the associated network change is applied. In this paper, we will not explore the fine-grained hierarchy of our dimensional data (for example, routers are naturally grouped into PoPs, which can be grouped into regions, which in turn can be grouped into countries), but the utility of this should be clear.

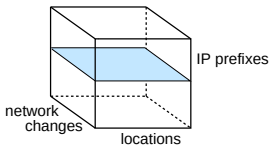

Figure 2: Data cube for network sensitivity analysis.

In this section, we study slices of the data cube that correspond to an IP prefix such as the one represented by the shaded surface in Figure 2. We define metrics of region-shift sensitivity computed over this surface.

4.2 Definitions

Let $\mathcal{U}$ be the universe set of vertices. Given an undirected weighted graph $G = (V, E, w, d)$, where $V \subset \mathcal{U}$, and a non-empty set of vertices $L \subset \mathcal{U}$, we say that L is the set of *root vertices*.[2] These root vertices represent the set of egress points for a destination prefix.

Every vertex $v \in \mathcal{U}$ has a local ranking of root vertices. The function $d(v, l)$ returns v's rank for root vertex l. When either v or $l \notin V$, $d(v, l) = \infty$. If $d(v, l) < d(v, l')$, then v is *closer* to l than to l'. The ranking function d represents the IGP distance between routers in a network, and consequently its computation depends on specific details of the intradomain routing protocol and the network configuration. In a network with no IGP hierarchy, $d(v, l)$ is the sum of the weights of the edges in a shortest path between v and l. If the network has a more complex IGP structure, then $d(v, l)$ can be computed using a model of the IGP hierarchy [5, 18].

Given a graph G with ranking function d and a root set L, each root vertex $l \in L$ induces a *region* $R \subseteq V$ such that:

$$R_l(G, L) = \left\{ v | \forall v \in V \text{ and } d(v, l) \leq d(v, l'), \forall l' \in L, l \neq l' \right\}.$$

This construction divides G into $|L|$ regions, and each region $R_l(G, L)$ is a shortest distance tree rooted in l. Note, though, that the division of G into regions is not a partition of the vertices in G because regions are not necessarily mutually disjoint. For instance, if $d(v, l) = d(v, l')$ for $l, l' \in L$, then $v \in R_l(G, L) \cap R_{l'}(G, L)$. We allow vertices to be in more than one region to model the potential of the tie-break decision (after the hot-potato step in the BGP decision process) to use any of the associated regions. Note that if $l \notin V$, then $R_l(G, L) = \emptyset$.

We define $RI(G, L, v) \subseteq L$ as the *region index set* of a vertex v in a graph G divided into regions according to the root set L.

$$RI(G, L, v) = \{ l | \forall l \in L, v \in R_l(G, L) \}.$$

The region index set of a vertex v is the set of root nodes that are closest to v. If L is the egress set for a destination prefix p, then $RI(G, L, v)$ represents v's entry in the routing matrix when considering p. If v is disconnected from the graph, then $\forall l \in L, d(v, l) = \infty$ and $RI(G, L, v) = \emptyset$.

For example, the graph G in Figure 3 presents the internal topology of the AS shown in Figure 1. The figure shows the division of this topology into regions for the destination prefix p from our previous example. The root set is the egress set for p, so $L = \{A, B\}$. All other nodes are internal nodes. An arrow from vertex i to vertex j represents that vertex j is the successor of vertex i in the shortest path to a root l. The distances from C to A and B are 9 and 10, respectively; therefore, C is in the region of A (i.e., it forwards packets using egress point A). The region $R_A = \{A, C, D, F, J\}$ represents all routers that forward packets to p using A as the egress point, and the region $R_B = \{G, J\}$ similarly represents the routers that use B as the egress. Note that router J is in both R_A and R_B, hence $RI(G, L, J) = \{A, B\}$.

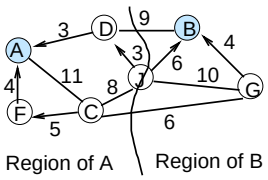

Figure 3: Example of the division of G into regions.

A *graph transformation* of G is a function $\delta : (V, E, w, d) \rightarrow (V', E', w', d)$ that deletes one or more edges or vertices, changes the weight of one or more edges, or adds one or more edges or vertices. Let ΔG be the class of graph transformations of G, such as the set of single edge deletions. Every graph transformation δ has a probability $\mathbb{P}(\delta)$ associated with it such that $\sum_{\delta \in \Delta G} \mathbb{P}(\delta) = 1$.

For each $\delta \in \Delta G$ and $v \in V$, we can compute the *region-shift function* $\mathcal{H}(G, L, v, \delta)$ as follows:

$$\mathcal{H}(G, L, v, \delta) = \begin{cases} 1, & \text{if } RI(G, L, v) \neq RI(\delta(G), L, v) \\ 0, & \text{otherwise} \end{cases}$$

Intuitively, the function $\mathcal{H}$ determines whether a vertex experiences a hot-potato routing change, i.e., whether a vertex v changes regions after a transformation δ is applied to G, thereby shifting region boundaries. Note that RI is a set of root vertices, hence the removal or addition of an element is considered a region shift. When a root $l \in L$ is deleted from G, it no longer belongs to any region and consequently the value of the region-shift function for l is one. For all other possible types of graph transformations δ, a root vertex l cannot change regions, therefore $\mathcal{H}(G, L, l, \delta) = 0$ for all $l \in L$.

Let us re-examine the example presented in Figure 3. Suppose that ΔG is the set of single edge deletions. Now consider the scenario where $\delta =$ *delete edge* CF. In an IP network, the deletion of an edge represents a link failure. Figure 4 presents the resulting

[2] We consider V a subset of a greater universe of vertices because we want to study cases in which vertices are deleted or added to the graph G.

graph $\delta(G)$ with the new division into regions. The distance of C to A changed from 9 to 11 while the distance of C to B remained 10. Hence, C changes from R_A to R_B and $\mathcal{H}(G, L, C, \delta) = 1$. All the other vertices are not affected by the change, so $\forall v \in V, v \neq C, \mathcal{H}(G, L, v, \delta) = 0$.

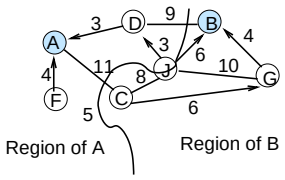

Figure 4: Example of the division of vertices into regions after deleting the edge CF.

4.3 Regional Sensitivity

To analyze a graph's sensitivity to graph transformations with respect to a set of root vertices, we construct a two-dimensional surface where there is a point for each pair (v, δ). The value of each point (v, δ) is set to $\mathcal{H}(G, L, v, \delta)$. Then, we introduce metrics that capture the sensitivity of a vertex to region shifts and the impact of a graph transformation on all the vertices in the graph by computing averages and maxima over one-dimensional slices of this plane as represented in Figure 5.

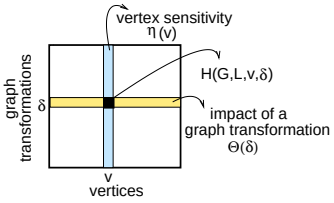

Figure 5: Regional sensitivity metrics are computed over a two-dimensional surface of vertices and graph transformations.

Vertex Sensitivity

The *vertex sensitivity* ($\eta \in [0, 1]$) describes the expected regional sensitivity of a vertex $v \in V$ given the set of graph transformations ΔG with probability $\mathbb{P}$.

$$\eta(G, L, \Delta G, \mathbb{P}, v) \quad = \quad \sum_{\delta \in \Delta G} \mathcal{H}(G, L, v, \delta).\mathbb{P}(\delta)$$

Vertex sensitivity captures the likelihood of a vertex to change regions when applying the class of graph transformations ΔG in G. In this section, we discuss sensitivity for a given set of parameters for a graph G, class of graph transformations ΔG, and graph transformation probability function $\mathbb{P}$. For conciseness of notation, we omit these parameters, hence $\eta(G, L, \Delta G, \mathbb{P}, v) \equiv \eta(L, v)$. We also define the average vertex sensitivity ($\hat{\eta}$), which is useful for evaluating the sensitivity of a graph.

$$\hat{\eta}(G, L, \Delta G, \mathbb{P}) \quad = \quad \frac{1}{|V|} \cdot \sum_{v \in V} \eta(G, L, \Delta G, \mathbb{P}, v)$$

Consider the graph G presented in Figure 6. Assume that all edges have the same weight, $L_1 = \{A, B\}$, ΔG is the class of single edge deletions, and all $\delta \in \Delta G$ are independent and have

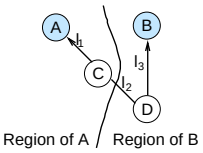

Figure 6: Example of a graph with vertices divided into regions as induced by $L_1 = \{A, B\}$.

equal probabilities. In this scenario, the four vertices are divided into two regions as illustrated in the figure.

Table 3 presents a cross-tabulation over the values of $\mathcal{H}(L_1, v, \delta)$ for all vertices in Figure 6. Each column δ_i corresponds to the deletion of the edge l_i. The last column presents the vertex sensitivity $\eta(L_1, v)$ for each vertex. Vertices A and B are roots, therefore no edge deletion can cause them to change regions and both $\eta(L_1, A)$ and $\eta(L_1, B)$ equal 0. Out of the three possible graph transformation, only the deletion of edge l_1 causes vertex C to change from R_A to R_B. Hence $\eta(L_1, C) = \frac{1}{3}$. The computation of D's sensitivity is similar. We will describe the last row of the table when we define the impact of a graph transformation and graph sensitivity.

	δ_1	δ_2	δ_3	$\eta(L_1, v)$
A	0	0	0	0
B	0	0	0	0
C	1	0	0	$\frac{1}{3}$
D	0	0	1	$\frac{1}{3}$
$\theta(L_1, \delta_i)$	$\frac{1}{4}$	0	$\frac{1}{4}$	$\sigma(L_1) = \frac{1}{6}$

Table 3: Sensitivity metrics using root set L_1 (Figure 6.

In general, values of $\eta(L, v)$ vary from 0 to 1. When $\eta(L, v)$ is close to 0, v is not sensitive to the class of graph transformations ΔG, i.e., it is unlikely that v changes regions given the division induced by L. By definition, $\eta(L, l) = 0$ for any root $l \in L$ when considering any class of graph transformations that does not include the deletion of root vertices. Routers that are in the set of egress points for a destination prefix always prefer routes learned from eBGP. Therefore, no intradomain changes (excluding the router crashing) can have an impact on route selection for such routers.

A value of $\eta(L, v) = 1$ means that v is sensitive to any graph transformation $\delta \in \Delta G$. Consider the graph presented in Figure 6 with only B as root ($L_2 = \{B\}$). Figure 7 shows the division in regions for this new scenario and Table 4 shows the values of the sensitivity metrics. In this example, A just has one route to reach the destination prefix p, thus any edge deletion disconnects A from p. Since the deletion of any edge in the graph disconnects A from the root B, $\eta(L_2, A) = 1$.

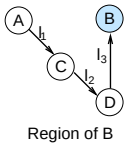

Figure 7: Example of a graph with root set $L_2 = \{B\}$.

Impact of a Graph Transformation

The *impact of a graph transformation* ($\theta \in [0, 1]$) determines the fraction of the vertices in V that experience region shifts after a

	δ_1	δ_2	δ_3	$\eta(L_2, v)$
A	1	1	1	1
B	0	0	0	0
C	0	1	1	$\frac{2}{3}$
D	0	0	1	$\frac{1}{3}$
$\theta(L_2, \delta_i)$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{3}{4}$	$\sigma(L_1) = \frac{1}{2}$

Table 4: Sensitivity metrics using root set L_2 (Figure 7.

given graph transformation.

$$\theta(G, L, \delta) = \frac{1}{|V|} \cdot \sum_{v \in V} \mathcal{H}(G, L, v, \delta)$$

We define a metric that captures the expected impact ($\hat{\theta}$) of the class of graph transformations ΔG with probability function $\mathbb{P}$:

$$\hat{\theta}(G, L, \Delta G, \mathbb{P}) = \sum_{\delta \in \Delta G} \theta(G, L, \delta).\mathbb{P}(\delta)$$

Let us revisit the example in Figure 6 to analyze the impact of edge deletions. The last row of Table 3 presents the values $\theta(L_1, \delta)$ for each edge deletion. The deletion of edge l_2 does not cause any vertex to change regions. This graph transformation has the least impact. Both the deletion of l_1 and l_3 impact one of the vertices in the graph, thus these transformations have a higher value of θ. The last row of Table 4 presents $\theta(L_2, \delta)$ for the scenario depicted in Figure 7. When only B is in the root set, the deletion of l_3 causes all other vertices to change regions, making $\theta(L_2, \delta_3) = \frac{3}{4}$.

As with vertex sensitivity, for any root set L and graph transformation δ, values of $\theta(L, \delta)$ vary between 0 and 1. $\theta(L, \delta) = 0$ represents a transformation that causes no region shifts in the graph, whereas $\theta(L, \delta) = 1$ happens when δ is a transformation that causes all the vertices in the graph to change regions. When the root set is composed of all vertices in the graph (i.e., $L = V$), then no graph transformation (with the exception of vertex deletions) causes a vertex to change regions. This configuration is not sensitive to changes in the graph and $\theta(L, \delta) = 0$. Note, however, that this scenario is extremely unlikely in an IP network. First, peering is not available everywhere. Second, the set of egress points for a prefix is usually much smaller than the number of routers in the network. On the other hand, if δ is the deletion of all $l \in L$, then all vertices $v \in V$ change regions and $\theta(L, \delta) = 1$. This other extreme is also unlikely for prefixes with more than one egress point.

Graph Sensitivity

Both the average vertex sensitivity $\hat{\eta}$ and the expected impact of a graph transformation $\hat{\theta}$ allow us to compare the overall region-shift sensitivity for $(G, L, \Delta G, \mathbb{P})$.

LEMMA 4.1.

$$\hat{\eta}(G, L, \Delta G, \mathbb{P}) = \hat{\theta}(G, L, \Delta G, \mathbb{P})$$

PROOF.

$$\begin{aligned}
\hat{\eta}(G, L, \Delta G, \mathbb{P}) &= \frac{1}{|V|} \cdot \sum_{v \in V} \eta(G, L, \Delta G, \mathbb{P}, v) \\
&= \frac{1}{|V|} \cdot \sum_{v \in V} \sum_{\delta \in \Delta G} \mathcal{H}(G, L, v, \delta).\mathbb{P}(\delta) \\
&= \sum_{\delta \in \Delta G} \frac{1}{|V|} \cdot \sum_{v \in V} \mathcal{H}(G, L, v, \delta).\mathbb{P}(\delta) \\
&= \sum_{\delta \in \Delta G} \theta(G, L, \delta).\mathbb{P}(\delta) \\
&= \hat{\theta}(G, L, \Delta G, \mathbb{P})
\end{aligned}$$

□

Therefore, we define *graph sensitivity* as

$$\begin{aligned}
\sigma(G, L, \Delta G, \mathbb{P}) &= \hat{\eta}(G, L, \Delta G, \mathbb{P}) \\
&= \hat{\theta}(G, L, \Delta G, \mathbb{P}).
\end{aligned}$$

We can now compute the graph sensitivity for the example presented in Figure 6. Over all the twelve possible v, δ combinations only two experience region changes and they have the same probability, so $\sigma(L_1) = \frac{1}{6}$. The scenario presented in Figure 7 is more likely to have region changes after an edge deletion. The graph sensitivity metric reflects this increased sensitivity. For L_2, $\sigma(L_2) = \frac{1}{2}$ and $\sigma(L_2) > \sigma(L_1)$.

When $\sigma(L) = 0$, the graph G with root set L is extremely robust to the graph transformations defined in ΔG. This happens when $L = V$ and ΔG does not include vertex deletions. Values of σ close to one arise when any graph transformation in ΔG causes all vertices to change regions. The only scenario in which σ is one is when ΔG contains only one transformation that is the deletion of all the roots together. In general, we expect the graph sensitivity to be $0 < \sigma(L) < 1$.

5. NETWORK SENSITIVITY

We now use the terminology introduced in the previous section to model hot-potato routing. Then, we present metrics that capture both control and data sensitivity to hot-potato changes.

5.1 From Regions to Hot Potatoes

We model an AS as a graph G, where vertices represent routers, edges are IP-level links, edge weights are the IGP cost associated with a link interface, and the IGP path costs (computed with or without IGP hierarchy) are incorporated by the ranking function d. The set of egress points for a destination prefix p is represented as a root set L_p. We assume that all routers $l \in L_p$ announce equally-good routes to reach p. This assumption implies that our model considers the set of best BGP routes after applying policies. In other words, we are only considering routes that are tied through step 5 in the BGP decision process (see Table 1).

A region R_l is the set of routers that use the route announced by egress point l to forward traffic to destination prefix p. A hot-potato change is a result of one or more intradomain path cost changes caused by some underlying event (such as a fiber cut, interface down, router crash, IGP weight change for maintenance, etc.). We model intradomain routing changes as a graph transformation. In this scenario, the region-shift function $\mathcal{H}$ determines whether a router experiences a hot-potato change for a destination prefix when a particular intradomain routing change happens.

One can think of vertex sensitivity as a metric that, given a network and a set of intradomain routing changes, determines the likelihood of a router to change its selection of egress point for a particular prefix. The impact of a graph transformation can be used to determine the fraction of routers that experience a hot-potato change after an intradomain routing change. The graph sensitivity metric σ reflects the overall impact of internal routing changes on the egress point that routers choose to forward packets to a destination prefix.

By the definition of regions, when a router is equidistant from two egress points, we consider it to be in two different regions at the same time. In practice, a router breaks this tie using a configuration-specific rule. For instance, by default Cisco routers prefer the older route [19], which leads to different decisions depending on the order of routing changes. A model that captures this tie-breaking rule would have to simulate routing dynamics. Even though there are some deterministic options for breaking this tie (such as comparing the ID of the egress points) that could potentially be incorporated into our model, the choice of the tie break is particular for each network configuration.

Instead, a network should be robust independent of routing dynamics (otherwise, the network design would be counting on luck!). The definition of region-shift function introduced in the previous section captures the worst-case sensitivity, because it considers that if there is any chance that a vertex v might change regions, then $\mathcal{H}(G, L, v, \delta) = 1$. We refer to this definition of region-shift function as $\mathcal{H}_{worst}$.

$$\mathcal{H}_{worst}(G, L, v, \delta) = \begin{cases} 1, & \text{if } RI(G, L, v) \neq RI(\delta(G), L, v) \\ 0, & \text{otherwise} \end{cases}$$

Because of the tie-breaking rule, when a node is in more than one region at the same time, some graph transformations may change the region index RI and yet cause no hot-potato change in practice. We define $\mathcal{H}_{best}$ as the best case scenario for the region-shift function. If there is any root vertex that belongs to $RI(G, L, v)$ and $RI(\delta(G), L, v)$, then we consider that v does not experience a hot-potato change.

$$\mathcal{H}_{best}(G, L, v, \delta) = \begin{cases} 1, & \text{if } RI(G, L, v) \cap RI(\delta(G), L, v) = \emptyset \\ 0, & \text{otherwise} \end{cases}$$

By computing the sensitivity metrics using $\mathcal{H}_{worst}$, we determine an upper bound for regional sensitivity. Similarly, using $\mathcal{H}_{best}$ determines a lower bound. In practice, the sensitivity metrics should lie between these two values depending on the tie-breaking rule adopted and the order of routing changes.

For clarity of notation, we define control plane and data plane sensitivity metrics using the region-shift function $\mathcal{H}$ instead of defining upper and lower bounds based on $\mathcal{H}_{worst}$ and $\mathcal{H}_{best}$. Our case study of the ISP network in Section 6 uses these definitions to compute upper and lower bounds of network sensitivity.

5.2 Control Plane Sensitivity

So far, our model has focused on the analysis of sensitivity when considering only one destination prefix. However, hot-potato disruptions result from a large number of routes changing simultaneously because of an intradomain routing change. Thus, we add another dimension to the data cube to capture the set of destination prefixes P and analyze the network state as a three-dimensional data cube (as the one presented in Figure 2) to understand the impact of a graph transformation on all the prefixes in the routing matrix.

Each destination prefix p has a set of egress points L_p. The set of egress points, however, is not unique per prefix. It is often the case that a number of destination prefixes share the same set of egress points. For example, the two networks presented in Figure 1 peer into two locations (peering routers A and B). In this example, all destination prefixes of customers of the neighbor AS share the egress set $\{A, B\}$. Let P be a set of destination prefixes, and β the mapping of prefixes to egress sets ($\beta : P \rightarrow 2^{\mathcal{U}}$.

The *routing-shift function* ($\mathcal{H}^{RM}(G, P, v, \delta)$) represents the fraction of destination prefixes in P for which there is a change in egress points after a graph transformation δ.

$$\mathcal{H}^{RM}(G, P, v, \delta) = \frac{1}{|P|} \cdot \sum_{p \in P} \mathcal{H}(G, \beta(p), v, \delta)$$

Routing sensitivity metrics are computed similarly to the region-shift metrics by replacing the region-shift function $\mathcal{H}$ with the routing-shift function $\mathcal{H}^{RM}$. One can view the routing-shift function as a transformation of the one-dimensional slice of the data cube for a pair (v, δ). After applying the transformation $\mathcal{H}^{RM}(G, P, v, \delta)$ for every pair (v, δ), we obtain a two-dimensional projection that is equivalent to the plane presented in Figure 5.

Node Routing Sensitivity

The *node routing sensitivity* ($\eta^{RM} \in [0, 1]$) describes the expected fraction of route shifts from a router's perspective. It represents the fraction of a router's BGP table that changes egress points.

$$\begin{aligned} \eta^{RM}(G, P, \Delta G, \mathbb{P}, v) &= \sum_{\delta \in \Delta G} \mathcal{H}^{RM}(G, P, v, \delta).\mathbb{P}(\delta) \\ \hat{\eta}^{RM}(G, P, \Delta G, \mathbb{P}) &= \frac{1}{|V|} \cdot \sum_{v \in V} \eta^{RM}(G, P, \Delta G, \mathbb{P}, v) \end{aligned}$$

Node routing sensitivity is useful for determining the routers in the network that are more susceptible to hot-potato changes. This metric does not differentiate between a router that experiences a few route shifts for most $\delta \in \Delta G$ and another that experiences rare but large route shifts. To differentiate between the two cases, we introduce η^{RM}_{max} to represent the worst case route shift for each node $v \in V$.

$$\eta^{RM}_{max}(G, P, \Delta G, v) = \max_{\delta \in \Delta G} \mathcal{H}^{RM}(G, P, v, \delta)$$

Node routing sensitivity η^{RM} can also be represented as the average vertex sensitivity across all prefixes as demonstrated by the following lemma.

LEMMA 5.1.

$$\eta^{RM}(G, P, \Delta G, \mathbb{P}, v) = \frac{1}{|P|} \cdot \sum_{p \in P} \eta(G, \beta(p), \Delta G, \mathbb{P}, v)$$

PROOF.

$$\begin{aligned} \eta^{RM}(G, P, \Delta G, \mathbb{P}, v) &= \sum_{\delta \in \Delta G} \mathcal{H}^{RM}(G, P, v, \delta).\mathbb{P}(\delta) \\ &= \sum_{\delta \in \Delta G} \frac{1}{|P|} \cdot \sum_{p \in P} \mathcal{H}(G, \beta(p), v, \delta).\mathbb{P}(\delta) \\ &= \frac{1}{|P|} \cdot \sum_{p \in P} \sum_{\delta \in \Delta G} \mathcal{H}(G, \beta(p), v, \delta).\mathbb{P}(\delta) \\ &= \frac{1}{|P|} \cdot \sum_{p \in P} \eta(G, \beta(p), \Delta G, \mathbb{P}, v) \end{aligned}$$

□

Consider again the graph presented in Figure 6. Assume that the number of prefixes $|P| = 10$ and that six prefixes use egress set L_1 (Figure 6) and remaining four use egress set L_2 (Figure 7). Table 5 presents the node routing sensitivity $\eta^{RM}(v)$ for each vertex in G computed using the values $\eta(L_1, v)$ and $\eta(L_2, v)$ presented in Tables 3 and 4, respectively. Since B is in both L_1 and L_2, no single edge deletion causes it to change regions with respect to L_1 or L_2. Hence, it is the least sensitive vertex in G; in fact, $\eta^{RM}_{max}(B) = 0$. Vertex A changes regions after any edge deletion when the root set is L_2, but never changes regions when the root set is L_1. Hence, we compute $\eta^{RM}(A) = \frac{0\times 6+1\times 4}{10} = 0.4$. Even though by examining η^{RM} vertex D is less sensitive than A and B, η^{RM}_{max} indicates that there is at least one graph transformation (in this case δ_3) for which all its routes change egresses.

	$\eta^{RM}(v)$	$\eta^{RM}_{max}(v)$
A	0.4	0.6
B	0	0
C	0.47	0.6
D	0.33	1

Table 5: Node routing sensitivity for the graph presented in Figure 6 with root sets L_1 and L_2.

Routing Impact of a Graph Transformation

The *routing impact of a graph transformation* ($\theta^{RM} \in [0, 1]$) represents the average fraction of BGP route shifts across all routers after an intradomain routing change.

$$\begin{aligned}\theta^{RM}(G, P, \delta) &= \frac{1}{|V|}.\sum_{v\in V}\mathcal{H}^{RM}(G, P, v, \delta)\\ \hat{\theta}^{RM}(G, P, \Delta G, \mathbb{P}) &= \sum_{\delta\in\Delta G}\theta^{RM}(G, P, \delta).\mathbb{P}(\delta)\end{aligned}$$

The routing impact of a graph transformation represents the average number of entries in the routing matrix that are changed because of a graph transformation. Similar to node routing sensitivity, we define a metric to represent the node that is most impacted by each graph transformation (θ^{RM}_{max}).

$$\theta^{RM}_{max}(G, P, \delta) = \max_{v\in V}\mathcal{H}^{RM}(G, P, v, \delta)$$

As with node routing sensitivity, the routing impact of a graph transformation can also be defined in terms of the impact of a graph transformation on each root set equivalence class.

LEMMA 5.2.

$$\theta^{RM}(G, P, \delta) = \frac{1}{|P|}.\sum_{p\in P}\theta(G, \beta(p), \delta)$$

We omit the proof because the steps are similar to that of Lemma 5.1.

Table 6 shows the routing impact of the deletion of each edge in G. The deletion of edge l_3 is the transformation with greatest impact. Even though it only impacts one vertex when considering root set L_1, it impacts all vertices for L_2. Indeed, we see from Table 6 that both $\theta^{RM}(\delta_3)$ and $\theta^{RM}_{max}(\delta_3)$ are higher than the impact of δ_1 and δ_2.

	$\theta^{RM}(\delta)$	$\theta^{RM}_{max}(\delta)$
δ_1	0.25	0.6
δ_2	0.2	0.4
δ_3	0.45	1

Table 6: Routing impact of single edge deletions on the graph presented in Figure 6 with root sets L_1 and L_2.

Overall Control Plane Sensitivity

The *overall control plane sensitivity* ($\sigma^{RM} \in [0, 1]$) is the average node routing sensitivity or the expected routing impact of a class of graph transformations. Control plane sensitivity represents the average fraction of the routing matrix that shifts in response to internal perturbations.

$$\begin{aligned}\sigma^{RM}(G, P, \Delta G, \mathbb{P}) &= \hat{\eta}^{RM}(G, P, \Delta G, \mathbb{P})\\ &= \hat{\theta}^{RM}(G, P, \Delta G, \mathbb{P})\end{aligned}$$

The overall control plane sensitivity for the example in Figure 6 with root sets L_1 and L_2 is $\sigma^{RM}(G, P, \Delta G, \mathbb{P}) = 0.3$. Overall control plane sensitivity provides an aggregated view of network sensitivity to IGP changes and can be used as one metric of comparing the robustness of the control plane of different network designs.

Perhaps more useful for determining the robustness of a network are the worst case node routing sensitivity (η^{RM}_{max}) and routing impact of graph transformations (θ^{RM}_{max}). High values of η^{RM}_{max} mean that there is at least one router in the network that has a high probability of experiencing hot-potato disruptions, which may then lead to router overload. Similarly, graph transformations with a high routing impact may lead to the overload of the control plane. A robust network should minimize σ^{RM}_{max}.

$$\begin{aligned}\sigma^{RM}_{max}(G, P, \Delta G, \mathbb{P}) &= \max_{v\in V}\eta^{RM}_{max}(G, P, \Delta G, v)\\ &= \max_{\delta\in\Delta G}\theta^{RM}_{max}(G, P, \delta)\end{aligned}$$

By identifying the most disruptive graph transformations, network operators can plan for them before maintenance activities or, longer term, add extra links or routers to reduce or avoid the most disruptive events. Practical constraints may prevent a network from being free of hot-potato disruptions. Knowledge of the areas of the network that are most vulnerable to hot-potato disruptions can be used when selecting the location to connect customers. For instance, it may be economically more advantageous to connect customers that use interactive applications such as voice and gaming in locations that are less susceptible to disruptions.

5.3 Data Plane Sensitivity

The previous section presented metrics to study variations in the routing matrix caused by IGP changes, i.e., the impact of hot-potato routing on the control plane. In this section, we combine routing with traffic demands and introduce metrics that measure the impact of hot-potato routing changes on the ingress-to-egress traffic matrix.

Let P be the set of destination prefixes and $\mathcal{I} \subseteq V$ be the set of ingress routers. $\mathcal{M}$ is an $|\mathcal{I}| \times |P|$ matrix, representing the ingress point to destination prefix traffic demand matrix. An element (v, p)

of $\mathcal{M}$ represents the volume of traffic from an ingress router $v \in \mathcal{I}$ to a destination prefix $p \in P$. We redefine the data cube from Figure 2 to study data plane sensitivity, each cell of the data cube presented in (v, δ, p) now contains the value $\mathcal{H}(G, \beta(p), v, \delta).\mathcal{M}(v, p)$, which represents the volume of traffic from a node v to a prefix p if v changes regions when δ is applied to G. Given the ingress-to-prefix demand matrix, the total inbound traffic at an ingress node v ($\mathbb{T}(v)$) is:

$$\mathbb{T}(v) = \sum_{p \in P} \mathcal{M}(v, p)$$

The *traffic-shift function* ($\mathcal{H}^{TM}(G, P, \mathcal{M}, v, \delta)$) represents the fraction of the traffic entering the network at ingress node v that switches egress points after the graph transformation δ.

$$\mathcal{H}^{TM}(G, P, \mathcal{M}, v, \delta) = \frac{1}{\mathbb{T}(v)} \cdot \sum_{p \in P} \mathcal{H}(G, \beta(p), v, \delta).\mathcal{M}(v, p)$$

The fraction of traffic that changes egresses may experience transient performance degradation during convergence and changes in forwarding path characteristics (such as congestion, longer RTTs, or packet filters).

We now define data plane sensitivity metrics as a function of the traffic-shift function.

Ingress Node Traffic Sensitivity

The *ingress node traffic sensitivity* ($\eta^{TM} \in [0, 1]$) describes the expected fraction of the traffic originating at ingress node v that switches egress points when considering the set of graph transformations in ΔG with probability $\mathbb{P}$. This metric captures the expected variation on v's entry of the traffic matrix for all graph transformations in ΔG. We also define η^{TM}_{max} to represent the largest traffic shift experienced by each node v when considering the set of graph transformations ΔG.

$$\begin{aligned}
\eta^{TM}(G, P, \Delta G, \mathbb{P}, \mathcal{M}, v) &= \sum_{\delta \in \Delta G} \mathcal{H}^{TM}(G, P, \mathcal{M}, v, \delta).\mathbb{P}(\delta) \\
\eta^{TM}_{max}(G, P, \Delta G, \mathcal{M}, v) &= \max_{\delta \in \Delta G} \mathcal{H}^{TM}(G, P, \mathcal{M}, v, \delta) \\
\hat{\eta}^{TM}(G, P, \Delta G, \mathbb{P}, \mathcal{M}) &= \frac{1}{|\mathcal{I}|} \cdot \sum_{v \in \mathcal{I}} \eta^{TM}(G, P, \Delta G, \mathbb{P}, \mathcal{M}, v)
\end{aligned}$$

η^{TM} can also be computed by averaging the total volume of traffic shifts across all root sets and all possible graph transformations over the total traffic originated at v. As discussed with routing sensitivity metrics, traffic sensitivity ranges from zero (indicating that traffic from an ingress point does not shift egress points when considering ΔG) to one (the other extreme, where any transformation in ΔG causes all the traffic originated at v to switch egress points).

Traffic Impact of a Graph Transformation

The *traffic impact of a graph transformation* ($\theta^{TM} \in [0, 1]$) represents the average across all ingress points of the fraction of the traffic that shifts because of a graph transformation δ. It captures the variation in the traffic matrix after the graph transformation δ.

$$\begin{aligned}
\theta^{TM}(G, P, \mathcal{M}, \delta) &= \frac{1}{|\mathcal{I}|} \cdot \sum_{v \in \mathcal{I}} \mathcal{H}^{TM}(G, P, \mathcal{M}, v, \delta) \\
\theta^{TM}_{max}(G, P, \mathcal{M}, \delta) &= \max_{v \in V} \mathcal{H}^{TM}(G, P, \mathcal{M}, v, \delta) \\
\hat{\theta}^{TM}(G, P, \Delta G, \mathbb{P}, \mathcal{M}) &= \sum_{\delta \in \Delta G} \theta^{TM}(G, P, \mathcal{M}, \delta).\mathbb{P}(\delta)
\end{aligned}$$

These metrics represent the fraction of the volume of traffic that shifts egress points due to an intradomain routing change. θ^{TM}_{max} represents the node that experiences the largest traffic shift because of a graph transformation. By computing this metric, network operators can make statements such as, "no single link failure will shift more than 1% of the traffic".

Overall Data Plane Sensitivity

The *overall data plane sensitivity* ($\sigma^{TM} \in [0, 1]$) describes the average ingress node traffic sensitivity or the expected traffic impact of a graph transformation. It captures the average change in the traffic matrix.

$$\begin{aligned}
\sigma^{TM}(G, P, \Delta G, \mathbb{P}, \mathcal{M}) &= \hat{\theta}^{TM}(G, P, \Delta G, \mathbb{P}, \mathcal{M}) \\
&= \hat{\eta}^{TM}(G, P, \Delta G, \mathbb{P}, \mathcal{M})
\end{aligned}$$

The maximum data plane sensitivity (σ^{TM}_{max}) represents the worst case traffic shift experienced by a node considering all graph transformations in ΔG.

$$\begin{aligned}
\sigma^{TM}_{max}(G, P, \Delta G, \mathcal{M}) &= \max_{v \in V} \eta^{TM}_{max}(G, P, \Delta G, \mathcal{M}, v) \\
&= \max_{\delta \in \Delta G} \theta^{TM}_{max}(G, P, \mathcal{M}, \delta)
\end{aligned}$$

Together the average and the maximum data plane sensitivity metrics allow us to compare different networks with respect to their robustness in the flow of traffic under intradomain routing changes. Network operators may also consider analyzing traffic for specific customers or applications separately. For customers using interactive applications any traffic shift may cause performance degradation. By studying data plane sensitivity for the subset of the addresses corresponding to those customers, either the network can be reprovisioned or the customer connectivity location can be changed to minimize disruptions due to internal events.

6. APPLYING THE MODEL

This section demonstrates the utility of our model and metrics by analyzing the sensitivity of a large AS of a tier 1 ISP network to link and router failures. First, we give a brief explanation of how to obtain the input parameters for the model from measurements collected from operational networks. Then, we analyze the control plane sensitivity of the tier 1 AS. An analysis of the data plane sensitivity of the network remains future work.

6.1 Obtaining Inputs for the Model

Most large ISPs routinely collect routing and traffic measurement data for network management purposes. One can leverage this data to extract the input parameters for our model as follows:

- The **network topology (G)** and the **ranking function (d)** can be derived either from snapshots of a router's IGP configuration or from archives of IGP routing messages collected by a route monitor.
- The **set of destination prefixes (P)** and **prefix-to-egress-set mapping (β)** can be computed by joining a collection of BGP tables or from archives of BGP routing messages.
- Snapshots of the **traffic demands $\mathcal{M}$** can either be estimated from link load statistics [12, 13] or can be measured directly at ingress routers [11, 20].

The ISP uses OSPF as its intradomain protocol and the network has been partitioned into several areas [2]. An OSPF monitor [18] deployed in the ISP network collects link-state advertisements from the network. The monitor is located in a Point of Presence (PoP) and has a direct physical connection to a router in area 0.[3] Thus, it receives detailed information about all links and routers in area 0, but only summarized information about routers in other areas. The summarized information consists of the distance to each router in the non-zero area from the border routers between area 0 and the non-zero area in question. We use the area 0 topology and the summarized information to construct snapshots of the network topology and to extract the ranking function d for each router in area 0. Note that even though we do not have detailed topology information for non-zero areas, the area 0 topology and summarized information allow us to compute the exact OSPF distance from any router in area 0 to any other router in the network.

The ISP uses an iBGP route-reflector hierarchy [10] inside the network for scalability reasons. A BGP monitor establishes iBGP sessions (running over TCP) to at least one route reflector per PoP (all these route reflectors belong to OSPF area 0). The monitor timestamps and archives all the BGP updates received over these sessions and dumps the routing table once a day to provide a periodic snapshot of the best route for each prefix. For each destination prefix p, we compute L_p as the union of all egress points selected by each of the route reflectors monitored.[4] Since we do not model the route-reflector hierarchy, we use the same mapping from prefix to egress set β for all routers in the network.

6.2 Case Study: An ISP Network

This section analyzes the control sensitivity of an AS in an ISP network. First, we present an in-depth analysis of control plane sensitivity that focuses on a recent snapshot of the AS collected on June 1, 2004. Then, we evaluate how control plane sensitivity evolves over time.

6.2.1 Control Plane Sensitivity Analysis

We illustrate how a network operator could use our model to analyze the control plane sensitivity of the network to internal routing changes. In the absence of accurate failure models for the network, network operators usually consider simple failure scenarios such as link and router failures. We consider two sets of graph transformations: $\Delta_L G$ is the set of all single link failures and $\Delta_R G$ is the set of all single router failures (excluding failures of the egress points). Assume that all failures within each set happen with equal probability. Because we only have detailed topology information for area 0 of the AS, we only consider failures in area 0 links and routers. Furthermore, we focus only on area 0 routers to compute control plane sensitivity.

We proceed with an exercise that shows how a hypothetical network operator could use our model to answer queries of different aspects of the network sensitivity to link and router failures.

How sensitive is the network to single link and single router failures?

Table 7 presents the overall control plane sensitivity of the network to single link and single router failures. We compute lower bound of control plane sensitivity (σ^{RM}) by using the best case routing shift function ($\mathcal{H}^{RM}_{best}$) as defined in Section 5.1. Similarly, the upper bound is computed using the worst case routing shift function ($\mathcal{H}^{RM}_{worst}$). For conciseness of notation, we denote control plane sensitivity to link failures as $\sigma^{RM}(\Delta_L G)$ and to router failures as $\sigma^{RM}(\Delta_R G)$. Both $\sigma^{RM}(\Delta_L G)$ and $\sigma^{RM}(\Delta_R G)$ are very close to zero, indicating that overall the network is quite robust to single link and router failures. The network is relatively more sensitive to router failures than link failures ($\sigma^{RM}(\Delta_L G) < \sigma^{RM}(\Delta_R G)$). This matches the intuition that the failure of a router should impact more paths than the failure of a single link. Note that σ^{RM} represents control plane sensitivity that is averaged over all routers and all failures; having a low value for both single link and router failures means that the network on average is robust to these set of failures. However, there may be some "outliers" from the average that could still perturb the network. The next question explores this issue further.

	Overall sensitivity (σ^{RM})	
	Lower Bound	*Upper Bound*
Single link failures ($\Delta_L G$)	0.00002	0.00010
Single router failures ($\Delta_R G$)	0.01001	0.01165

Table 7: Overall control plane sensitivity of the AS.

What is the largest disruption that can happen in the network?

Table 8 presents the worst case control plane sensitivity σ^{RM}_{max}. Recall that σ^{RM}_{max} represents the pair (v, δ) that has the maximum routing sensitivity in the network. The upper-bound sensitivity to both $\Delta_L G$ and $\Delta_R G$ is almost 1. This indicates that there is at least one router that is extremely sensitive to at least one of the link and router failures. Assume that v_{max}, δ_{max} represents the worst case scenario. Values of $\sigma^{RM}_{max} = 1$ means that v_{max} shifts egresses for *all* destination prefixes as a consequence of the graph transformation δ_{max}. This leads to the conclusion that, although on average the network is robust to single link and router failures, there exists a set of routers that are extremely sensitive to certain failures.

	Worst case sensitivity (σ^{RM}_{max})	
	Lower Bound	*Upper Bound*
Single link failures ($\Delta_L G$)	0.5586	0.9974
Single router failures ($\Delta_R G$)	0.9974	0.9974

Table 8: Overall control plane sensitivity of the AS.

Our hypothetical operator digs deeper by slicing the data cube per failure and computing the impact of individual failures, and then slicing it per router and computing each router's routing sensitivity.

Which failures are most disruptive?

Figure 8 shows the routing impact (θ^{RM}) of link and router failures. The top of the bar in the plot is the upper bound sensitivity computed using $\mathcal{H}^{RM}_{worst}$ as defined in Section 5.1, the bottom is the lower bound computed using $\mathcal{H}^{RM}_{best}$, and the middle point is the average of the two (we use this notation on all the remaining plots in this case study). The x-axis is the fraction of graph transformations sorted according to the upper-bound routing impact. The average routing impact of both link and router failures is low. For instance, $\theta^{RM} < 0.01$ for 93% of link failures and 59% of router failures. It is clear from this plot that there are a few failures that have a considerable impact on some routers. In particular, some router failures

[3] OSPF areas form a hub-and-spoke topology such that area 0 forms the hub and non-zero areas form the spokes.

[4] There may be routes learned at some border routers that are not announced internally. In this situation no other router in the network can use these border routers as egress points, so it will not impact our sensitivity measurements.

have $\theta^{RM} > 0.2$, which means that they impact an average of 20% of the destination prefixes across all routers. Although the impact of link failures is lower, there may be some routers that experience large route shifts because of some link failures. After determining the most disruptive failures, the operator can use this information while making decisions on traffic engineering and network provisioning. The next step is to understand which routers are sensitive to hot-potato disruptions.

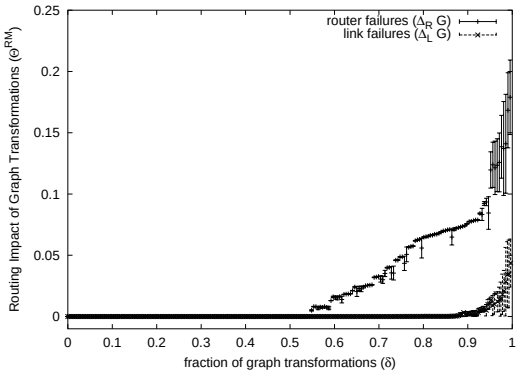

Figure 8: Routing impact of router and link failures.

Which routers are most sensitive?

Figure 9 shows the upper and lower bounds for node routing sensitivity to link and router failures. The x-axis is the fraction of all routers in area 0 sorted according to their upper-bound routing sensitivity. Almost all the routers experience very few routing changes on average. The average sensitivity may not be the best indicator: a router that experiences small route shifts for a number of failures may have the same node routing sensitivity (η^{RM}) as another router that experiences a very large route shift for only one of the failures. The latter case is arguably more disruptive than the former.

It is also interesting to note that there is a high variance among routers. A router's sensitivity to internal changes depends on its location relative to the closest and second-closest egress points for most destination prefixes. This variance of routing sensitivity across routers is consistent with the empirical findings presented in [4]. Indeed, if we rank routers based on the number of hot-potato changes as measured by the algorithm presented in [4], and rank the same routers according to their node routing sensitivity to single link failures, we find that the ranking is the same. For single router failures, however, the ranks do not agree. Although this might sound counter-intuitive, our metrics depend on the failure model we use. Given that router failures are rare events in practice, it is not surprising that the empirical results are more consistent with single link failure sensitivity metrics.

We observe that node routing sensitivity on average is very low, which is not surprising since we consider all failures to be equally probable and η^{RM} represents the average sensitivity over all possible failures. Next, the operator investigates node routing sensitivity further by analyzing the worst case routing shift for each router.

What is the largest routing shift experienced by each router?

Figures 10 and 11 present the worst case node routing sensitivity η^{RM}_{max} for single router failures and for single link failures, respectively. Over 30% of the routers experience considerable hot-potato

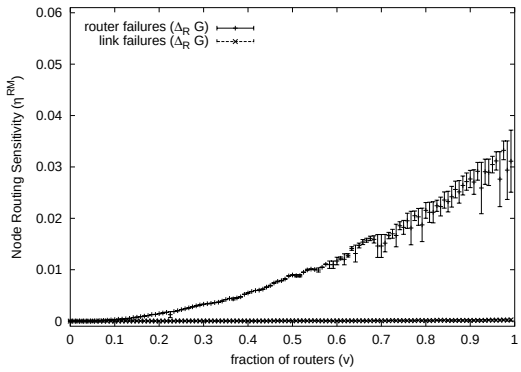

Figure 9: Node routing sensitivity to router and link failures.

disruptions for at least one of the router failures.[5] However, the impact of a particular router failure is usually limited to a few routers (usually located in the same PoP). Fewer routers experience hot-potato disruptions caused by link failures when compared to those caused by router failures.

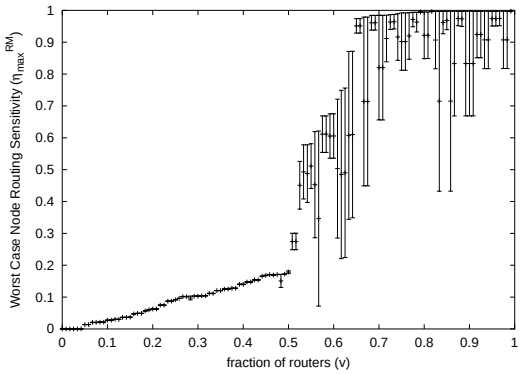

Figure 10: Worst case node routing sensitivity to router failures.

It is important to note the wide gap between lower and upper bounds for some nodes in Figures 10 and 11. This suggests that a substantial fraction of routing shifts for each router depends on the non-deterministic tie breaking step in the BGP decision process.

Note that the order of the routers in Figure 10 is different than that presented in Figure 11. Comparing node routing sensitivity to router failures with that of link failures for each router v, we find that 49% of the routers are more sensitive to link failures than router failures. Although counter-intuitive at first, such scenarios may arise in practice because hot-potato changes only occur when a graph transformation (link or router failure) changes the relative distance from the router to the closest and second-closest egress points. Consider the example presented in Figure 12, where we assume that A and B are egress points for all destination prefixes. Node E will shift all its routes from egress A to B upon the failure of link AC, whereas the failure of node C does not cause any shift. This indicates that optimizing the network topology or configuration to minimize sensitivity to one type of graph transformations may result in an increase in the sensitivity to some other type of transformation.

[5]Note that many of these routing changes may be unavoidable without overloading the links leading to the old egress points. We discuss these trade-offs in Section 6.2.3.

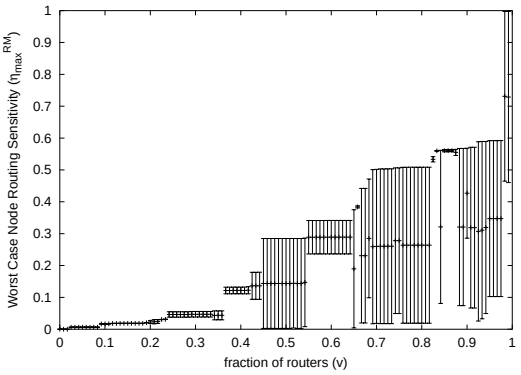

Figure 11: Worst case node routing sensitivity to link failures.

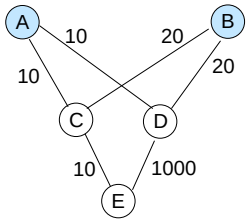

Figure 12: Example showing higher sensitivity to a single link failure than a single router failure.

We analyze the most sensitive routers further. We call the router with highest $\eta_{max}^{RM}(\Delta_R G)$ (rightmost router in Figure 10) router A and the router with highest $\eta_{max}^{RM}(\Delta_L G)$ router B. Figures 13 and 14 present the distribution of route shifts ($\mathcal{H}^{RM}$) for routers A and B, respectively. Only a very small fraction of the failures cause the worst case hot-potato disruption. Overall, even the most sensitive routers are robust to most changes in the network; only a small fraction of failures cause large routing shifts.

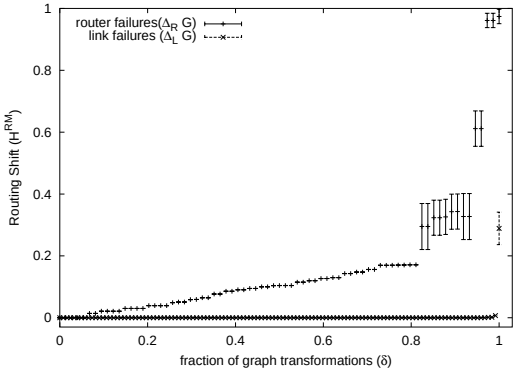

Figure 13: Distribution of control plane sensitivity of router A.

Network operators can use the knowledge of which routers are more sensitive to hot-potato disruptions when deciding in which location to connect customers. For instance, customers that use interactive applications such as VoIP or gaming are more sensitive to the forwarding instabilities caused by a hot-potato change. An ISP may decide that connecting a customer in a less sensitive location may be worth the cost of a long-haul link.

6.2.2 Temporal Variation of Sensitivity

The sensitivity analysis presented in the previous section focused

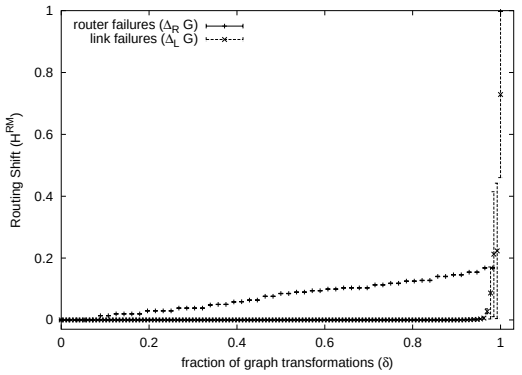

Figure 14: Distribution of control plane sensitivity of router B.

on one snapshot of the network state. A large tier 1 ISP, however, has hundreds of routers and links and consequently the state of network is in constant flux due to failures and maintenance activities. This makes the selection of a representative snapshot of the network particularly challenging. We now study one snapshot of (G, P, β) per month from February 2003 to June 2004 to determine the sensitivity of our analysis to the choice of the network snapshot and the variation of control plane sensitivity over time.

Figure 15 presents the overall control plane sensitivity σ^{RM} to both $\Delta_L G$ and $\Delta_R G$ for one snapshot per month during this 17-month period. The network's overall control plane sensitivity to both types of failures is low. As seen earlier, the network is relatively more sensitive to router failures (from 0.012 to 0.021) than link failures (from 0.00003 to 0.0001). We also observe that there is no dramatic variation of σ^{RM} during this 17-month period, just a small decrease between months one and two, another between eight and nine, and a decrease in the last month. This decrease in overall control plane sensitivity indicates that on average the network has become more robust to router and link failures over time.

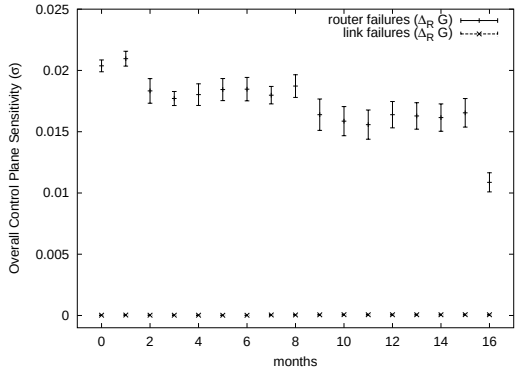

Figure 15: Overall sensitivity to router and link failures over time.

6.2.3 Discussion

Our analysis of the control plane sensitivity shows that, on average, the ISP network is very robust to link and router failures. Nevertheless, there is room for improvement: some link and router failures can cause routers to shift egress points for a large number of destination prefixes. After identifying the most sensitive routers and the most disruptive failures, network operators and designers can use this information to improve network robustness. In doing

so, there are some design guidelines and operational practices that should be considered to minimize the impact of internal routing changes:

Link and node redundancy. One approach to minimize sensitivity is to replicate all paths in the network. Clearly, this approach is too expensive in practice. However, our analysis shows that there are a few critical links and routers. Replicating these network components can help minimize hot-potato disruptions. Our model can be used iteratively to determine the network sensitivity after the addition of new components.

Selection of peering locations. Another approach to minimize overall sensitivity is to have only one egress point or to have peering at every router. Neither of these solutions is desirable for practical reasons: (i) peering locations depend on business relations and it is not feasible for an ISP to peer with all other ISPs in every location; and (ii) selecting only one peering location for each destination prefix is not desirable from a reliability and traffic engineering perspective. ISPs can use the knowledge of which locations of the network are more sensitive to disruptions and prioritize adding connections to peers at these locations.

Reconfiguring the network. Selecting the best configuration of link weights to reduce network sensitivity adds an extra dimension to the problem of optimizing link weights for traffic engineering. Although optimizing link weights for one type of graph transformations may increase sensitivity for other types, more accurate failure models can help guide our analysis. Reconfiguration of link weights can be used to avoiding hot-potato disruptions during planned maintenance activities. In this case, operators know in advance which graph transformation will be applied to the graph and when. They can use our model to identify a less disruptive configuration of the link weights to be deployed before the event.

It should be clear from this discussion that all of these factors represent a trade-off that network designers and operators need to consider when managing their networks. Our sensitivity analysis can assist in making these trade-offs.

7. CONCLUSIONS

In this paper, we develop methods for characterizing network sensitivity to intradomain routing changes, or *hot-potato disruptions*, to ultimately improve network robustness. First, we propose and describe an analytic model of the interaction between intra and interdomain routing and its impact on both the control and data planes of an ISP network. Based on this model, we define a set of metrics for describing a network's sensitivity to intradomain routing perturbations. We study control plane sensitivity of a large AS of a tier 1 ISP to link and router failures. This analysis demonstrates the utility of our model for identifying which routers are particularly sensitive to internal perturbations and which failures would cause most disruptions.

As future work, we plan to improve the accuracy of the model by incorporating iBGP hierarchies. We also plan to derive realistic failure models and study both control and data plane sensitivity for these models. Finally, we would like to use the model as a basis of a tool that network operators can use for improving the robustness of their networks.

Our approach for building more robust networks has focused on improving network design given routing protocols as they exist today. Network robustness problems are intrinsic in the way BGP reacts to small IGP cost changes. Newer routing protocols should be designed with the goal of network-wide robustness in mind.

Acknowledgments

We would like to thank Jennifer Rexford for her priceless encouragement and feedback at every stage of this work. AT&T supported Renata Teixeira's internships and she thanks AT&T research for its help and support, especially Albert Greenberg. Renata was also supported in part by a fellowship from Capes/Brazil and by AT&T support of the UCSD Center for Networked Systems. Voelker was supported in part by DARPA FTN Contract N66001-01-1-8933. Jay Borkenhagen provided invaluable guidance in our understanding of the backbone network. Thanks also to Christophe Diot, Alex Snoeren, our shepherd Anja Feldmann, and the anonymous reviewers for their helpful comments.

8. REFERENCES

[1] Y. Rekhter and T. Li, "A Border Gateway Protocol 4 (BGP-4)." RFC 1771, March 1995.

[2] J. Moy, "OSPF Version 2." RFC 2328, April 1998.

[3] R. Callon, "Use of OSI IS–IS for Routing in TCP/IP and Dual Environments." RFC1195, December 1990.

[4] R. Teixeira, A. Shaikh, T. Griffin, and J. Rexford, "Dynamics of Hot-Potato Routing in IP Networks," in *Proc. ACM SIGMETRICS*, June 2004.

[5] A. Feldmann, A. Greenberg, C. Lund, N. Reingold, and J. Rexford, "NetScope: Traffic Engineering for IP Networks," *IEEE Network Magazine*, pp. 11–19, March 2000.

[6] B. Fortz, J. Rexford, and M. Thorup, "Traffic Engineering with Traditional IP Routing Protocols," *IEEE Communication Magazine*, October 2002.

[7] B. Fortz and M. Thorup, "Optimizing OSPF/IS-IS Weights in a Changing World," *IEEE J. Selected Areas in Communications*, vol. 20, no. 4, pp. 756 – 767, 2002.

[8] N. Feamster, J. Winick, and J. Rexford, "A Model of BGP Routing for Network Engineering," in *Proc. ACM SIGMETRICS*, June 2004.

[9] S. Agarwal, C.-N. Chuah, S. Bhattacharyya, and C. Diot, "Impact of BGP Dynamics on Intra-Domain Traffic," in *Proc. ACM SIGMETRICS*, June 2004.

[10] S. Halabi and D. McPherson, *Internet Routing Architectures*. Cisco Press, second ed., 2001.

[11] 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 Trans. Networking*, vol. 9, June 2001.

[12] A. Medina, N. Taft, K. Salamatian, and C. Diot, "A Taxonomy of IP Traffic Matrix Estimation: Existing Techiniques and New Directions," in *Proc. ACM SIGCOMM*, August 2002.

[13] Y. Zhang, M. Roughan, C. Lund, and D. Donoho, "An Information-Theoretic Approach to Traffic Matrix Estimation," in *Proc. ACM SIGCOMM*, August 2003.

[14] G. Wilfong and T. G. Griffin, "On the Correctness of IBGP Configuration," in *Proc. ACM SIGCOMM*, August 2002.

[15] J. Gray, A. Bosworth, A. Layman, and H. Pirahesh, "Data cube: a relational aggregation operator generalizing group-by, cross-tabs and subtotals," No. 1, pp. 29–53, 1997.

[16] S. Chaudhuri and U. Dayal, "An overview of data warehousing and OLAP technology," *ACM SIGMOD Record*, vol. 26, March 1997.

[17] A. Gupta, V. Harinarayan, and D. Quass, "Aggregate query processing in data warehousing environments," in *VLDB*, pp. 358–369, 1995.

[18] A. Shaikh and A. Greenberg, "OSPF Monitoring: Architecture, Design and Deployment Experience," in *Proc. USENIX Symposium on Networked Systems Design and Implementation*, March 2004.

[19] "BGP Best Path Selection Algorithm." `http://www.cisco.com/warp/public/459/25.shtml`.

[20] A. Lakhina, K. Papagiannaki, M. Crovella, C. Diot, E. Kolaczyk, and N. Taft, "Structural analysis of network traffic flows," in *Proc. ACM SIGMETRICS*, June 2004.