Toward Valley-Free Inter-domain Routing

Sophie Y. Qiu, Patrick D. McDaniel*, and Fabian Monrose
Dept. of CS, Johns Hopkins University *Dept. of CSE, Pennsylvania State University
{yuqiu,fabian}@cs.jhu.edu mcdaniel@cse.psu.edu

***Abstract*— ASes in inter-domain routing receive little information about the quality of the routes they receive. This lack of information can lead to inefficient and even incorrect routing. In this paper, we quantitatively characterize BGP announcements that violate the so-called valley-free property—an indicator that universal best practices are not being preserved in the propagation of routes. Our analysis indicates that valley announcements are more pervasive than expected. Approximately ten thousand valley announcements appear every day and involve a substantial number of prefixes. 11% of provider ASes propagate valley announcements, with a majority of violations happening at intermediate providers. We find that large surges of violating announcements can be attributed to transient configuration errors. We further propose a dynamic mechanism that provides route propagation information as transitive attributes of BGP. This information implicitly reflects the policies of the ASes along the path, without revealing the relationship of each AS pair. BGP-speaking routers use this information to identify (and presumably avoid) routes that violate the valley-free property.**

I. Introduction

Creating and using policy in inter-domain routing is hard. Understanding when a route should be selected as "best" is a function of the current network state, relationships between routing bodies, and other administrative and operational factors. However, routers have only a partial (and surprisingly small) view of this information—which often leads to poor and incorrect decisions. These poor decisions translate to poor performance, unbalanced loads, and network instability. Policy is the means by which network administrators correct for a lack of information by tuning the rules for the processing of routes.

Policy is also one of the major factors that lead to BGP's bewildering complexity. Independent of routing mechanism (e.g., path vector, link-state, etc.) and operation details, many BGP problems result from inability to satisfy policy requirements [1]. For example, it has been shown that interaction of independently implemented policies may lead to policy disputes and cause BGP to oscillate indefinitely [2], [3]. Gao [4] argues that BGP export rules indicate that AS paths should be *valley-free*, i.e., typically ASes want to filter and avoid propagating routes that use a small (customer) AS to transit between two larger ASes [1]. However, it has been found that some advertised AS paths do not conform to the valley-free property [5], [6], [7]. In response, Feamster *et al.* [1] point out that it is not sound to assume that ASes advertise routes correctly in the first place. They further suggest that detecting routes that violate policy remains a daunting problem in inter-domain Routing.

Unfortunately, guaranteeing that BGP routes are indeed valley-free and globally reasonable is particularly difficult because of the lack of strict guidelines on setting policies and the property that BGP keeps policy information private. In fact, it is suggested that static examination not only requires a global view of policies but is also NP-complete [3]. Therefore, a practical solution must be dynamic. One such approach is to extend BGP and allow policy conflicts to be identified at run time [8]. There have also been some calls for applying guidelines, which capitalize on AS commercial relationships, in configuring their routing policies and for disclosing (part of) policy-related information [9], [10].

In this paper, we quantitatively characterize the extent to which BGP routes violate the valley-free property. We further propose a dynamic mechanism that extends BGP and enables ASes to avoid constructing and propagating valley routes. We begin our study with empirical analysis of real-world BGP traffic. Our results suggest that valley announcements are more pervasive than expected. Approximately ten thousand valley announcements appear every day and surprisingly a substantial percentage of prefixes (e.g., as high as 26% during a one month period) are involved. Moreover, large surges of valley announcements due to configuration errors also occur and significantly increase the routing load. Furthermore, we characterize valley patterns, observe their dynamics, examine the contributors, and explore potential causes for these valleys.

Lastly, we evaluate effective solutions to guard against valley routes. We propose a dynamic mechanism that adds additional information to BGP which implicitly reflects the policies of the ASes along the path, without revealing the relationship of each AS pair. In particular, we associate the AS path with different *states*. The state symbolizes the pattern of the advertised AS path by reflecting the sequence of edges the path has traversed. The state information helps ASes decide whether to import or export the route. If the route is to be further propagated, the AS updates the state accordingly based

on the previous state and its agreements with neighbor ASes. This way, construction and propagation of routes that are not valley-free can be prevented. Furthermore, we show that adopting the mechanism and eliminating the valley paths does not affect connectivity.

The remainder of this paper is structured as follows. Section II presents the background knowledge about AS relationships, routing policies and the valley-free property. In Section III, we empirically analyze real-world BGP traffic and quantitatively characterize BGP announcements that are not valley-free. In Section IV, we propose a dynamic mechanism that extends BGP and effectively prevents construction and propagation of valley routes. Related work is presented in Section V, and we conclude in Section VI.

II. Background

At a high level, the Internet consists of a collection of interconnected, independently operated networks referred to as Autonomous Systems. Each AS represents a portion of the network under single administrative control. Routing between ASes is constrained by the commercial agreements between administrative domains, which translate into routing policies.

The commercial agreements between ASes can be categorized into customer-provider, peering, mutual-transit, and mutual-backup agreements [11]. A provider provides Internet connectivity to its customers. A pair of peers exchange traffic between their respective customers. Mutual-transit agreements allow administrative domains (normally small ISPs close to each other) to connect to the Internet through each other. In case the connection to its provider fails, an AS may get connectivity to the Internet via another AS through mutual-backup agreement. The commercial agreements among ASes play an important role in deciding how the traffic flows in the Internet and are thus critical for us to understand the Internet structure and end-to-end performance. However, such information is not publicly available due to privacy concerns or commercial reasons.

In general, routing policies include import and export policies. Import policies specify whether to deny or permit a received route and assign a local preference indicating how favorable the route is. Export policies allow ASes to determine whether to propagate their best routes to the neighbors. Most ASes follow the following guidelines in their export policy settings [11], [4]: while exporting to a provider or peer, an AS can export its routes and its customer routes, but usually does not export its provider or peer routes; while exporting to a customer or sibling, an AS can export its routes and its customer routes, as well as its provider or peer routes. These exporting rules indicate that AS path should be *valley-free, i.e., after a provider-to-customer edge or a peer-to-peer edge, the AS path can not traverse customer-to-provider edges or another peer-to-peer edge* [4]. In other words, a provider-to-customer or peer-to-peer edge can only be followed by provider-to-customer or sibling-to-sibling edges. However, it has been observed that some advertised AS paths do not conform to the valley-free property [5], [6], [7], implying that common policies are not followed in those cases.

Figure 1 shows an example that illustrates AS topology and relationships. For example, paths $\{D,A,E\}$ (assuming the leftmost AS is the originating AS), $\{A,E,F\}$,$\{D,A,B\}$ and $\{H,E,F,G\}$ are valley-free while paths $\{D,A,E,B,G\}$, $\{A,E,F,G\}$, $\{D,A,B,C\}$, and $\{H,E,F,G,C\}$ are not.

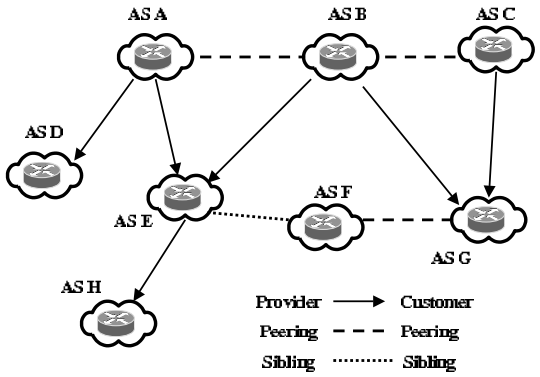

Fig. 1. AS topology and relationship example

The adverse effects of valley routes (i.e., the routes that are not valley-free) include but are not limited to: (1) unnecessarily increasing routing load by introducing valley announcements that should not appear; (2) inadvertently violating commercial agreements in operation; (3) intended routes not being chosen because of valley ones; (4) ASes being unaware of unintentionally transiting traffic over valley paths; (5) obtaining Internet connectivity that is provided for others.

III. Characterizing Valley Violations

We begin by analyzing real-world BGP updates that violate the valley-free property, implying that common BGP policies are not complied with during the propagation of such advertisements. Because information regarding AS relationships is not publicly available, we adopt Gao's work [4] on heuristically inferring AS relationships. Based on BGP routing table snapshots captured at Routeviews [12], we build an AS topology graph where nodes represent ASes. We then apply Gao's heuristics and label the edges according to the relationships of the end nodes, i.e., *provider-to-customer*, *customer-to-provider*, *peer-to-peer*, and *sibling-to-sibling*. We examine BGP updates archived at Routeviews for a period across several months (Apr–Jul 2006), and quantitatively

characterize those BGP announcements that are not valley-free.

A. Observing Valley Announcements

Valleys		Announcements	Distinct paths	Prefixes
Apr	#	458,498	42,067	22,209
	%	0.15%	1.01%	9.93%
May	#	487,458	49,474	17,400
	%	0.16%	1.20%	7.90%
Jun	#	2,155,565	65,035	60,237
	%	0.60%	1.43%	26.4%
Jul	#	393,332	46,162	52,490
	%	0.12%	1.00%	23.0%

TABLE I
VALLEYS OBSERVED IN BGP UPDATES

Table I shows the number of BGP announcements that contain valleys (i.e., are not valley-free) observed in each month, as well as the numbers of distinct AS paths and the prefixes that are affected (i.e., that are involved in valley announcements). Although the percentage of valley announcements is relatively low, it does not suggest that valleys need not receive significant attention. In fact, there are hundreds of millions of BGP announcements in each month and even a small percentage can be substantial (e.g., 2 million in June). Additionally, an overwhelming majority of BGP announcements reflect prefix oscillations (repetitive prefix re-additions and subsequent withdrawals) [13], which makes the percentage of other abnormalities appear trivial and ignored. Indeed, surprisingly a substantial percentage of prefixes (e.g., as high as 26% in June) have been affected by valleys.

Figure 2 illustrates the CDF of the valley announcements observed from 46 different observation points (i.e., peers of Routeviews' server). It appears that a high percentage of valley announcements are observed through a relatively small number of peers, e.g., 70% valley announcements through 10 peers in June.

Figure 3 illustrates the number of valley announcements observed each day during the four-month period. Approximately ten thousand valley announcements appear every day. Moreover, a few large surges which significantly increased routing load can be seen from the figure, especially in the month of June. For example, as many as about 0.7 million valley announcements were observed during a one-day period. We return to an examination of these surges in Section III-D.

B. Characterizing Valley Patterns

To better understand their characteristics, we further examine the patterns of the valleys. Based on how the valley-free property is violated, we classify valley violations into four categories:

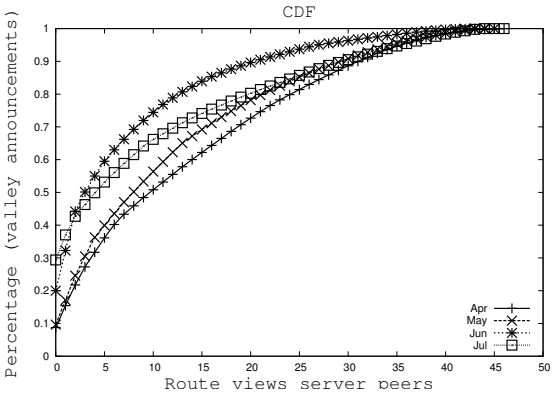

Fig. 2. CDF of the valley announcements observed from 46 Routeviews' peers

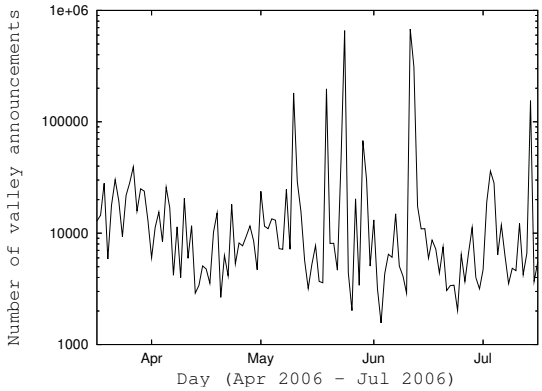

Fig. 3. Valley announcements in Apr - Jul, 2006

- *Type I:* Customer-provider edge is observed after provider-customer edge, i.e., an AS propagates a path that has traversed a provider-customer edge to its provider.
- *Type II:* Customer-provider edge is observed after peer-peer edge, i.e., an AS propagates a path that has traversed a peer-peer edge to its provider.
- *Type III:* Peer-peer edge is observed after provider-customer edge, i.e., an AS propagates a path that has traversed a provider-customer edge to its peer.
- *Type IV:* Peer-peer edge is observed after peer-peer edge, i.e., an AS propagates a path that has traversed a peer-peer edge to its peer.

Table II shows the number and percentage of different types of valley violations observed in each month. Note that the total number of valley violations is higher than that of valley announcements (e.g., 647,073 violations versus 487,458 valley announcements in May), because in some cases one announcement contains more than one valley violation. The results indicate that a majority of valleys are *Type I* or *III* violations, while the other

Violation		*Type I*	*Type II*	*Type III*	*Type IV*
Apr	#	490,046	4,097	184,421	3,667
	%	71.83%	0.60%	27.03%	0.54%
May	#	524,304	6,384	113,813	2,572
	%	81.03%	0.98%	17.59%	0.40%
Jun	#	2,224,941	464	533,603	771
	%	80.6%	1.68e-4	19.3%	2.79e-4
Jul	#	241,076	7,49	246,788	1,741
	%	49.16%	0.15%	50.33%	0.36%

TABLE II
BREAKDOWN OF VALLEY VIOLATIONS BASED ON PATTERNS

two types (*Type II* and *IV*) are relatively rare. Therefore, it appears that ASes violate valley-free property more frequently when propagating advertisements received from providers than when propagating advertisements received from peers. It is also not surprising that *Type I* valley is the dominating type since provider-customer relationships are much more common than sibling and peering relationships [4].

The *valley-free* property suggests that, after a provider-to-customer or peer-to-peer edge, the advertised AS path should not traverse customer-to-provider or another peer-to-peer edge. When the property is not maintained, we refer to the edge which the AS path should not traverse as *violation edge*, and refer to the last provider-to-customer or peer-to-peer edge before the violation edge as *critical edge*. We use violation edge and critical edge to identify a *valley*. A particular valley may appear in different valley paths and in multiple valley announcements. For example, assume both ASes A and C are AS B's providers. If the sequence of ASes [1] ..., A, B, C, ... appear in an advertised AS path, we call the edge $B-A$ the violation edge, and $C-B$ the critical edge. These two edges define a particular *valley* and B is the AS responsible for the violation.

Table III shows the number of distinct valleys and the number of ASes that propagate valley routes. We cluster valley violations (and distinct valleys) based on the contributing ASes. The results indicate that a relatively small number of ASes propagate a high percentage of valley routes. For example, as illustrated in Figure 4 that shows the CDF of the valley violations (and distinct valleys) contributed by the 410 ASes in May 2006, the top 50 contributing ASes are responsible for 90% of all violations (about 80% distinct valleys).

C. Valley Dynamics

To explore the dynamics of valley announcements, we compare BGP updates of different months, and examine the valley paths that are repeatedly announced. As shown in Table IV, a relatively small percentage (8-16%) of advertised valley paths are re-announced in the next month.

[1]Here we assume that the rightmost AS is the originating AS based on the format of BGP update messages.

	Distinct valleys	Contributing ASes
Apr	2,752	405
May	3,299	410
Jun	3,452	440
Jul	3,447	430
Apr–Jul	8,749	771

TABLE III
DISTINCT VALLEYS AND THE CONTRIBUTING ASES

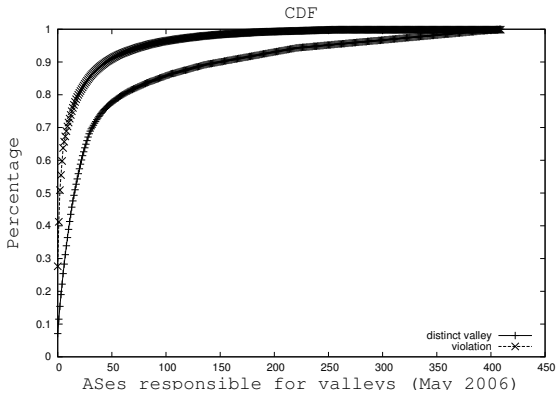

Fig. 4. CDF of the valleys contributed by 410 ASes in May 2006

Valley path	Re-announced in next month	
	#	%
Apr (42,067)	6,733	16.01%
May (49,474)	3,948	7.98%
Jun (65,035)	5,108	7.85%
Jul (46,162)	4,014	8.70%

TABLE IV
DYNAMICS OF VALLEY PATHS.

During the four-month observation period, we have observed 8,749 distinct valleys (identified by the violation edge and critical edge) in BGP updates. From the perspective of distinct valley, 70% of valleys appear to be transient—they disappear within one month. Here we use a relatively long period as the threshold because in most cases, valley routes caused by export misconfigurations do not affect connectivity directly and these errors are not easily noticed [7]. As a result, they take a relatively long time to be corrected. Another 5.8% of distinct valleys are persistently propagated during the four-month period, and the remaining valleys are observed intermittently.

D. Valley Contributors and Causes

Recent work [14] classifies the Internet infrastructure into three hierarchy levels: the *core* forms the top level which consists of tier-1 providers, the *middle* includes those intermediate ASes that provide transit services, and the *edge* consists of stub ASes that are customers only. Under such classification, a total of 15 ASes are considered as core, 6,580 as middle, and 19,158 as edge. We adopt this classification, and examine the number of ASes of different hierarchy levels that contribute to valley routes. It appears that a majority of the contributing ASes (732 out of 771) are middle ASes, and the remaining contributors include 25 edge ASes and 14 core ASes. It is not surprising that most core ASes are involved in valley routes because they are responsible for transiting a significant volume of traffic in the Internet. However, it is interesting to notice that AS 4513 (Globix Corporation) is the only top level AS that did not propagate any valley announcements during the entire observation period. Overall, 11% (746 out of 6,595) of provider ASes (core or middle ASes) are observed propagating valley announcements from time to time, while customer ASes appear to comply with the valley-free property very closely (only 0.1%, 25 out of 19,158, make violations).

Potential causes for valley violations can be unusual policies of some ASes, exceptions, misconfigurations, or intentional violations. However, it appears that a significant portion of valley announcements are caused by misconfigurations (errors). For example, AS 19429 (ETB-Colombia), the top 1 culprit AS for the large surges observed in Figure 3, is responsible for 697,266 valley announcements (3,385 distinct valley paths) of June 2006. It exported routes from its provider AS 1239 (Sprint) to another provider AS 3491 (Beyond The Network America, Inc.) and therefore caused a large number of *Type I* valley violations. In July 2006, only 66 announcements (25 distinct AS paths) are observed containing this valley, indicating that the problem is largely fixed. Other top contributing ASes for the surges in Figure 3, such as AS 26656 (Misys Intl. Banking Sys., Inc.), AS 19262 (Verizon), and AS 24103 (Greenfield-AS-AP), are each involved in several hundreds of thousands of valley announcements (thousands of distinct valley paths) in June. ASes 26656 and 24103 violated the valley-free property by exporting routes from one provider to another provider, and AS 19262 exported routes from one provider to its six peers. However, no announcements that contain these valleys are observed in July, implying that those valleys are very likely due to transient configuration errors in June.

Misconfigurations can arise for a variety of reasons and are yet to be well understood [7]. Frequently misconfigurations arise because of human factors such as inadvertent administrative errors. Also, router initialization or filtering may be incorrectly implemented. Moreover, configuration-related databases are found to be obsolete or not consistent [15]. Additionally, a poor understanding of configuration and policy semantics might be the likely reason for these misconfigurations.

In some cases the propagation of valley routes appears deliberate. A few ASes (mostly middle-size intermediate providers) do not seem to follow common export policies, or they tend to have more *flexible* policies of their own.They are involved in tens or hundreds of distinct valleys which consist of different valley types, and persistently propagate valley announcements during the observation period. It is possible that these ASes have more complex commercial agreements with their neighbors, or that special operational or commercial strategies are employed in such cases.

AS relationship inference: Undoubtedly, our experimental results are affected by the accuracy of the heuristics [4] we adopted for inferring AS relationships. Incorrectly classified AS relationships may cause either valley-free routes to be labeled as valleys, or conversely valley violations not being detected. However, Gao's results show that 99% of their inference between AT&T and its neighboring ASes are confirmed by AT&T internal information [4]. Additionally, recent work on evaluating AS relationship inference [6] suggests that Gao's algorithm achieves 94% overall accuracy and 99% accuracy for provider-customer type relationship classification. Moreover, the volume and frequency of observed valley announcements (a majority of which are *Type I*) indicate that they cannot be explained away by small errors in classification. Hence, while it can be said that reported results are quantitatively affected by these errors, they are very unlikely to be qualitatively affected.

It remains difficult to accurately characterize AS relationships. For example, AS relationships may be dynamic (while uncommon) due to administrative organization changes. During connectivity failures, some ASes may exploit hidden transient backup relationships. There are also complex instances that AS relationships are defined at prefix level, instead of AS level. For all these reasons and cases, improving the algorithms for assessing relationships and the data upon which they are based will certainly increase the accuracy of our analysis. We plan to exploit these improvements as they become available.

IV. Protection Mechanism

The common BGP export rules suggest that AS path should be valley-free [4]. Typically an AS wants to filter and avoid propagating routes that have valleys—those that use a small (customer) AS to transit between two larger ASes [1]. However, our empirical study in Section

III shows that approximately ten thousand valley announcements appear every day and involve a substantial percentage of prefixes. Moreover, large surges of valley announcements which significantly increase routing load due to configuration errors have been observed. In this section, we explore effective solutions to guard against valley routes.

The challenge of detecting valley routes is that an AS must know the relationships between other ASes. However, most ASes are reluctant to share this information. Our solution is to extend BGP with information that *implicitly* reflects the policies of the ASes along the path, without revealing details about their business relationships. Specifically, we propose to add *state* to the routing protocol that reflects the pattern of the advertised AS path. The state is dynamically associated to the AS path as a transitive BGP attribute. Through examining the current state of a received route and its relationships with neighbor ASes, an AS can prevent constructing and propagating routes that are not valley-free, and thus avoid violating common BGP export policy. The state implicitly reflects the relationships of the ASes along the path without revealing the exact relationship of each AS pair.

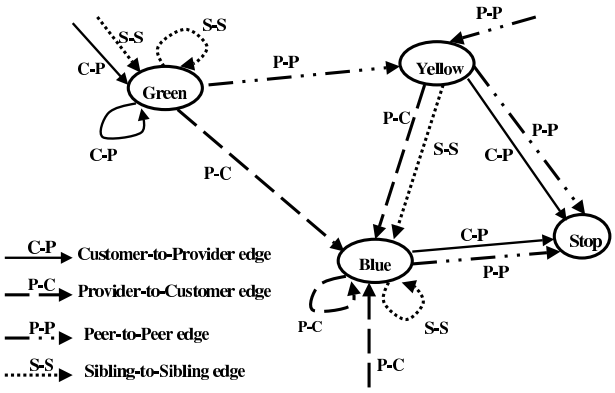

Fig. 5. State transition of AS paths

Figure 5 illustrates the transition of the states of advertised AS paths and the basic algorithm is summarized in Algorithm 1. During the propagation process of the path, the state is updated based on the previous state and the policy of each AS along the path. The state symbolizes the pattern of the advertised AS path by reflecting the sequence of edges the path has traversed. *Green* implies that all edges traversed so far are either customer-to-provider or sibling-to-sibling edges, so it is safe to export the route to anybody. *Yellow* indicates that the last edge of the advertised path is peer-to-peer edge, while other edges are customer-to-provider or sibling-to-sibling edges. *Blue* indicates that the advertised path has traversed either one provider-to-customer or peer-to-peer edge, and the last edge is either provider-to-customer or sibling-to-sibling edge. Therefore, if the received path is *Blue* or *Yellow*, an AS knows it has traversed at least one provider-to-customer or peer-to-peer edge, and will not further announce the route to its provider or peer. With such additional information, construction and propagation of valley routes can be effectively avoided.

Algorithm 1 Associate *state* to the advertised AS path

```
/* AS A_i advertises a prefix to AS A_{i+1} */
if Relation(A_i, A_{i+1}) = "Customer→Provider" or "Sibling→Sibling" then
    route r ← {{A_i}, Green};
if Relation(A_i, A_{i+1}) = "Peer→Peer" then
    route r ← {{A_i}, Yellow};
if Relation(A_i, A_{i+1}) = "Provider→Customer" then
    route r ← {{A_i}, Blue};

/* AS A_i receives a route r = {{A_{i-1}, ..., A_0}, State} from A_{i-1}. A_i updates the route should it be further propagated to A_{i+1} */
if Relation(A_i, A_{i+1}) = "Customer→Provider" then
    if State = "Green" then
        route r ← {{A_i, A_{i-1}, ..., A_0}, Green};
    else
        stop
else if Relation(A_i, A_{i+1}) = "Provider→Customer" then
    route r ← {{A_i, A_{i-1}, ..., A_0}, Blue};
else if Relation(A_i, A_{i+1}) = "Sibling→Sibling" then
    if State = "Green" then
        route r ← {{A_i, A_{i-1}, ..., A_0}, Green};
    else if State = "Yellow" or State = "Blue" then
        route r ← {{A_i, A_{i-1}, ..., A_0}, Blue};
else if Relation(A_i, A_{i+1}) = "Peer→Peer" then
    if State = "Green" then
        route r ← {{A_i, A_{i-1}, ..., A_0}, Yellow};
    else
        stop
```

A few questions may arise regarding the proposed mechanism. First, one may wonder whether eliminating the valley routes affects connectivity. To answer this question, we extract the valley paths observed in Section III and examine the connectivity between the end ASes, i.e., the originating AS and the last AS. For example, for each of the affected 29,216 AS pairs (the two end ASes of the valley path) in June 2006, we simulate route propagation on the annotated partially-directed AS graph starting from the origin AS. We follow our proposed mechanism such that the propagated AS paths are guaranteed to be *valley-free*. Our experiment results show that there always exist non-valley paths connecting the pairs (100% success rate). Therefore, eliminating

valley routes should not affect connectivity.

Secondly, although most ASes obey common BGP export polices, some ASes may have more flexible policies of their own and choose to construct and advertise valley routes. We leave the question whether such practices should be encouraged to future work. However, it is advisable that such routes are labeled and other ASes receiving such routes are informed. For example, the protocol (see Algorithm 1) can be revised such that, when an AS *intentionally* constructs a valley path it labels the state of the route as *red*. Upon receiving such route, other ASes are aware that it is a valley route and can decide whether to accept it at their will. If they further propagate the route, the red state is passed along. This way, an AS can express its preference not to follow common export policies for these routes and other ASes are informed.

V. Related Work

Many BGP problems result from the inability to satisfy policy requirements [1]. BGP policies are implemented locally with little global knowledge. Varadhan *et al.* [2] showed that interaction of independently defined policies may cause routing problems such as persistent oscillations. Griffin *et al.* suggested that the static analysis approach to solve policy dispute requires full knowledge of the AS graph and the routing policies of each AS. Unfortunately such information is unavailable and the situation is not expected to change any time soon. Moreover, the complexity of static checking is NP-complete [8], [3]. However, Gao and Rexford [9] observed that, if ASes apply a set of guidelines that capitalize on AS commercial relationships while configuring their route import policies, then BGP is provably shown to converge. The guidelines suggest that routing via a customer is preferred over routing via a peer or a provider and backup routes have the lowest preference. They further proposed a routing registry that requires each AS to disclose its relationships with neighbor ASes. Recent work on the next generation Inter-domain routing protocol [10] also advocates explicitly publishing the provider-customer relationships and restricting the normal paths to those that obey the hierarchies defined by these relationships. Our proposed mechanism extends BGP with dynamic information which implicitly reflects the relationships of the ASes along the path, and guarantees that common BGP export policies are complied with. A number of other solutions are not aimed at dealing directly with policy-compliance, but with the authenticity and freshness of the BGP advertisements [16], [17], [18], [19], [20], [21], [22], [23].

AS relationships play an important role in deciding how the traffic flows and are thus vital to understand Internet infrastructure. In recent years several algorithms [4], [24], [5], [6] have been proposed to infer AS relationships. For example, Gao [4] categorized AS relationships into provider-to-customer, customer-to-provider, peer-to-peer, and sibling-to-sibling relationships, based on the observation that a provider is typically larger than its customers and two peers are of comparable size. Subramanina *et al.* [24] formally defined the problem and presented an approach combining AS paths from multiple vantage points. Xia *et al.* [6] propose to retrieve partial relationships from BGP community, AS-SET objects, and routing policies recorded in IRR database and apply such partial information in inference.

The network community has also started trying to infer BGP policies and understand their implications. Wang *et al.* [25] studied both the BGP import and export policies network operators employ to configure their networks and how factors such as traffic engineering impact these policies. Most ASes follow certain rules in their export policy settings [11], [4]. These common BGP export policies suggest that AS path should be *valley-free* [4]. However, recent work [5], [6] on AS relationships has observed that valley paths exist in routing tables and pointed out that these paths are likely causing the inference algorithms to be inaccurate. Our work quantitatively characterizes valley BGP announcements for a four-month period from a variety of aspects and further examines their patterns, dynamics, contributors, and causes. Recent work on misconfigurations [7] showed that origin and export misconfigurations were indeed pervasive and further investigated their causes through email surveys. Their study suggests that reducing administrative mistakes by minimizing operator interaction and promoting self-configured systems remains a high priority task.

VI. Conclusion

In this paper we characterized the BGP announcements that violate the valley-free property on the Internet. This analysis shows that valley announcements are more pervasive than one might expect: approximately ten thousand valley announcements appear every day and (over time) involve a substantial percentage of the advertised prefixes (e.g., 26% in a one-month period). Further, surges of valley announcements due to configuration errors have been observed, and 11% of provider ASes (core or middle ASes) propagate valley announcements regularly, with a majority of violations happening at intermediate providers (middle ASes). All of this goes to paint a rather clear picture: route valleys occur with high frequency, are emitted from many sources, and effect many routes. This suggests that suppression of such paths would have a potentially large positive impact on the quality and stability of Internet routing.

We address valley advertisements by proposing a dynamic mechanism that adds path-state information to

advertisements. This transitive attribute attached to each advertised path implicitly reflects the policies of the ASes along the path without revealing the relationships among the ASes in the path. Such global information makes it possible to prevent constructing and propagating undesirable routes, and will lead to the suppression of potentially damaging valley routes.

VII. Acknowledgments

This work is supported in part by NSF grant SCI-0334108.

References

[1] N. Feamster, H. Balakrishnan, and J. Rexford, "Some foundational problems in Interdomain routing," in *ACM SIGCOMM Workshop on Hot Topics in Networks (HotNets)*, pp. 41–46, 2004.

[2] K. Varadhan, R. Govindan, and D. Estrin, "Persistent route oscillations in Inter-domain routing," *Computer Networks, 32:1-16*, 2000.

[3] T. Griffin, F. Shepherd, and G. Wilfong, "The stable paths problem and Inter-domain routing," *IEEE/ACM Trans. Networking, 10:232-243*, 2002.

[4] L. Gao, "On inferring autonomous system relationships in the Internet," *IEEE/ACM Trans. on Networking*, vol. 9, no. 6, pp. 733–745, 2000.

[5] G. Battista, M. Patrignani, and M. Pizzonia, "Computing the types of the relationships between autonomous systems," in *Proc. of INFOCOM*, pp. 156–165, 2003.

[6] J. Xia and L. Gao, "On the evaluation of AS relationship inferences," in *Proc. of GLOBECOM*, pp. 1373–1377, 2004.

[7] R. Mahajan, D. Wetherall, and T. Anderson, "Understanding BGP misconfiguration," in *Proc. of SIGCOMM*, pp. 3–16, 2002.

[8] T. Griffin and G. Wilfong, "An analysis of BGP convergence properties," in *Proc. of SIGCOMM*, pp. 277–288, 1999.

[9] L. Gao and J. Rexford, "Stable Internet routing without global coordination," in *Proc. of SIGMETRICS*, pp. 307–317, 2000.

[10] L. Subramanian, M. Caesar, C. Ee, M. Handley, M. Mao, S. Shenker, and I. Stoica, "HLP: A next generation inter-domain routing protocol," in *Proc. of SIGCOMM*, pp. 13–24, 2005.

[11] G. Huston, "Interconnection, peering and settlements," *Internet Protocol Journal*, vol. 2, pp. 2–23, June 1999.

[12] *University of Oregon Route View Project*. `http://routeviews.org`.

[13] S. Qiu, P. McDaniel, F. Monrose, and A. Rubin, "Characterizing address use structure and stability of origin advertisement in Inter-domain routing," in *Proc. of IEEE Symposium on Computers and Communications*, pp. 489–496, 2006.

[14] R. Oliveira, B. Zhang, D. Pei, R. Izhak-Ratzin, and L. Zhang, "Quantifying path exploration in the Internet," in *Proc. of IMC*, pp. 269–282, 2006.

[15] G. Siganos and M. Faloutsos, "Analyzing BGP policies: Methodology and tool," in *Proc. of INFOCOM*, pp. 1640–1651, 2004.

[16] S. Kent, C. Lynn, and K. Seo, "Secure Border Gateway Protocol (Secure-BGP)," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 582–592, April 2000.

[17] J. Ng, "Extensions to BGP to support secure origin BGP," in *Network Working Group*, 2003.

[18] W. Aiello, J. Ioannidis, and P. McDaniel, "Origin authentication in Inter-domain routing," in *Proc. of ACM Conference on Computer and Communications Security*, pp. 165–178, 2003.

[19] Y. Hu, A. Perrig, and M. Sirbu, "SPV: Secure path vector routing for securing BGP," in *Proc. of SIGCOMM*, pp. 179–192, 2004.

[20] L. Subramanian, V. Roth, I. Stoica, S. Shenker, and R. Katz, "Listen and Whisper: Security mechanisms for BGP," in *Proc. of NSDI*, pp. 127–140, 2004.

[21] M. Lad, D. Massey, D. Pei, Y. Wu, B. Zhang, and L. Zhang, "PHAS: A prefix hijack alert system," in *Proc. of USENIX Security*, pp. 153–166, 2006.

[22] J. Karlin, S. Forrest, and J. Rexford, "Pretty good BGP: Protecting BGP by cautiously selecting routes," in *Proc. of IEEE ICNP*, 2006.

[23] S. Qiu, F. Monrose, A. Terzis, and P. McDaniel, "Efficient techniques for detecting false origin advertisements in Inter-domain routing," in *Proc. of IEEE ICNP Secure Network Protocols (NPsec)*, pp. 12–19, 2006.

[24] L. Subramanian, S. Agarwal, J. Rexford, and R. H. Katz, "Characterizing the Internet hierarchy from multiple vantage points," in *Proc. of INFOCOM*, pp. 618–627, 2002.

[25] F. Wang and L. Gao, "Inferring and characterizing Internet routing policies," in *Proc. of IMC*, pp. 15–26, 2003.