Robustness of Class-Based Path-Vector Systems*

Aaron D. Jaggard† Vijay Ramachandran‡

July 21, 2004

Abstract

Griffin, Jaggard, and Ramachandran introduced in [4] a framework for understanding the design principles of path-vector protocols such as the Border Gateway Protocol (BGP), which is used for inter-domain routing on the Internet. They described as an application of their framework a study of Hierarchical-BGP-like systems where routing at a node is determined by the relationship with the next-hop node on a path (*e.g.*, an ISP-peering relationship) and some additional scoping rules (*e.g.*, the use of backup routes). These systems are called *class-based path-vector systems*. The robustness of these systems depends on the presence of a global constraint on the system, but an adequate constraint has not yet been given. In this paper, we give the best-known sufficient constraint that guarantees robust convergence. We show how to generate this constraint from the design specification of the path-vector system. We also give centralized and distributed algorithms to enforce this constraint, discuss applications of these algorithms, and compare them to algorithms given in previous work on path-vector protocol design.

1 Introduction

The standard Internet inter-domain routing protocol, the Border Gateway Protocol (BGP), determines routes using independently configured policies in autonomously administered networks. Little global coordination of policies takes place between domains, or autonomous systems (ASes), because (1) ASes are reluctant to reveal details about internal routing configuration, and (2) BGP contains no reliable mechanism to permanently attach information to a route as it is shared throughout the network. However, without global coordination, interaction of locally configured policies can lead to global routing anomalies [1, 5, 8, 10], *e.g.*, route oscillation and inconsistent recovery from link failures.

*This work was partially supported by the U.S. Department of Defense (DoD) University Research Initiative (URI) program administered by the Office of Naval Research (ONR).

†Dept. of Mathematics, Tulane University, New Orleans, LA, USA. adj@math.tulane.edu. Partially supported by ONR Grant N00014–01–1–0795 and by an NSF VIGRE Postdoc.

‡Dept. of Computer Science, Yale University, New Haven, CT, USA. vijayr@cs.yale.edu. Partially supported by a 2001–2004 DoD National Defense Science and Engineering Graduate (NDSEG) Fellowship, by ONR Grant N00014–01–1–0795, and by NSF Grant ITR–0219018.

Because the techniques used to configure policies and the protocol's specification have evolved separately, there is an inherent trade-off between, on one hand, maintaining rich semantic expressiveness available with current vendor-developed configuration languages and autonomy in policy configuration, and, on the other hand, guaranteeing that the protocol will converge robustly, *i.e.*, predictably, even in the presence of link and node failures. Griffin, Jaggard, and Ramachandran [4] showed that achieving all three of these design goals requires a non-trivial global constraint on the network, but they left open the question of how to identify and enforce the constraint.

This paper answers this question in the context of class-based path-vector systems. Path-vector systems, introduced in [4], provide a formal model for design and analysis of path-vector protocols and their policy-configuration languages. Class-based systems focus on a generalization of next-hop-preference routing, where routing policy for an AS can be defined by the relationships (commercial or otherwise) between it and its neighboring ASes. The canonical example of such a system is a simplified version of BGP that takes into account the economic realities of today's commercial Internet—that ASes are connected to their customers, providers, and peers and that there are preference guidelines used to decide between routes learned from neighbors of different classes. The scope of class-based systems, however, goes beyond this "Hierarchical BGP" system; the framework can also be used to build and analyze systems with complete autonomy and those that allow arbitrary next-hop preference routing. Furthermore, any protocol specification that can be described by a countable-weight, monotone path-vector algebra [9] can also be described by a class-based path-vector system [7].

In this paper, we provide the best known robustness constraint for class-based systems. The constraint ensures the robustness of networks that satisfy it; it is in fact the best possible robustness guarantee because, in networks that do not satisfy it, some set of nodes may write policies that cause route oscillations. (Our proof of this constructs such policies.) We give an algorithm to generate the constraint given only the design specification of the system. We then provide centralized and distributed algorithms to check networks for violation of the constraint and discuss their applications, including how to use our results to check a network with arbitrary next-hop preferences for potential bad interactions. The distributed algorithm reveals almost no private policy information, provides several options for correcting a constraint violation, and has constant message complexity per link and limited storage at each node. We compare and contrast our algorithms with those in previous work.

Although it may be sufficient to provide a supplementary protocol enforcing some global conditions (and, indeed, the distributed algorithm in this paper, modified for BGP, can be run alongside BGP to detect potentially bad policy interactions), there are several benefits to this approach of analyzing robust protocol convergence from a design-framework perspective. First, the algorithms in our paper preclude all policy-based oscillations in advance; as long as the constraint is enforced, the protocol can safely run on any network. Second, the approach is an integral part of designing policy-configuration languages.

The design framework identifies provably sufficient local and global conditions needed for a protocol to achieve its design goals. Our paper precisely gives the trade-off between the strength of local policy guidelines built into the policy-configuration language and the strength of the global assumption needed in the broad class-based context. The designer can use these results to consider what balance between local and global enforcement is desired and can incorporate the guidelines generated by the results in this paper into the design of multiple high-level policy languages—all before running the protocol on an actual network.

1.1 Related Work

Several papers have presented policy-configuration guidelines or design principles for path-vector routing so that global routing anomalies can be avoided. Gao and Rexford [3] showed that route preferences and scoping consistent with the Hierarchical-BGP example mentioned above give stable path-vector routing without global coordination. Griffin, Shepherd, and Wilfong in [5] defined an abstraction of path-vector routing and identified a sufficient condition for local policies that prevents policy-induced route oscillation, and Griffin and Wilfong in [6] used those conditions to give a simple path-vector routing algorithm that dynamically detects policy-induced route oscillations. In addition, Gao, Griffin, and Rexford in [2] combine the results from these papers to modify a simplified version of BGP to perform stable back-up routing.

Building on this work, the papers by Griffin, Jaggard, and Ramachandran [4] and Sobrinho [9] developed formal models for path-vector routing so that properties of path-vector protocols can be studied without involving details of protocol dynamics or of actual networks. Both papers prove results about protocol convergence based on the design specification of the protocol itself. They present an application of their frameworks that generalizes systems like Hierarchical BGP (mentioned above), incorporating the policy guidelines from [2, 3, 5] into the design of protocol systems. This paper completes the analysis of this application: It answers the questions left open in [4] and gives results that can be used in the general design of protocols and policy-configuration languages.

References

[1] Cisco Field Note. Endless BGP Convergence Problem in Cisco IOS Software Releases. Oct. 2001. `http://www.cisco.com/warp/public/770/fn12942.html`

[2] L. Gao, T. G. Griffin, and J. Rexford. Inherently Safe Backup Routing with BGP. In *Proc. INFOCOM 2001*.

[3] L. Gao and J. Rexford. Stable Internet Routing Without Global Coordination. *ACM/IEEE Trans. on Networking*, 9(6): 681–692, 2001.

[4] T. G. Griffin, A. D. Jaggard, and V. Ramachandran. Design Principles of Policy Languages for Path-Vector Protocols. In *Proc. ACM SIGCOMM'03*, pp. 61–72,

Aug. 2003. Extended version: Yale Univ. Tech. Report 1250 (YALEU/DCS/TR-1250), Apr. 2004, `ftp://ftp.cs.yale.edu/pub/TR/tr1250.{ps,pdf}`.

[5] T. G. Griffin, F. B. Shepherd, and G. Wilfong. The Stable Paths Problem and Interdomain Routing. *ACM/IEEE Trans. on Networking*, 10(2):232–243, 2002.

[6] T. G. Griffin and G. Wilfong. An Analysis of BGP Convergence Properties. In *Proc. ACM SIGCOMM'99*, Sept. 1999.

[7] A. D. Jaggard and V. Ramachandran. Relating Two Formal Models of Path-Vector Routing. Manuscript, Jul. 2004.

[8] D. McPherson, V. Gill, D. Walton, and A. Retana. BGP Persistent Route Oscillation Condition. RFC 3345, Aug. 2002.

[9] J. L. Sobrinho. Network Routing with Path Vector Protocols: Theory and Applications. In *Proc. ACM SIGCOMM'03*, pp. 49–60, Aug. 2003.

[10] K. Varadhan, R. Govindan, and D. Estrin. Persistent Route Oscillations in Interdomain Routing. *Computer Networks*, 32:1–16, 2000.