7-2012

Breif Announcement: A Calculus of Policy-Based Routing Systems

Anduo Wang
University of Pennsylvania

Carolyn Talcott
SRI

Alexander J. T. Gurney
University of Pennsylvania

Boon Thau Loo
University of Pennsylvania, boonloo@cis.upenn.edu

Andre Scedrov
University of Pennsylvania

Wang, A., Talcott, C., Gurney, A., Loo, B., & Scedrov, A., Brief Announcement: A Calculus of Policy-Based Routing Systems, ACM Symposium on Principles of Distributed Computing, July 2012, doi: 10.1145/2332432.2332501

Breif Announcement: A Calculus of Policy-Based Routing Systems

Abstract
The BGP (Border Gateway Protocol) is the single inter-domain routing protocol that enables network operators within each autonomous system (AS) to influence routing decisions by independently setting local policies on route filtering and selection. This independence leads to fragile networking and makes analysis of policy configurations very complex. To aid the systematic and efficient study of the policy configuration space, this paper presents a reduction calculus on policy-based routing systems. In the calculus, we provide two types of reduction rules that transform policy configurations by merging duplicate and complementary router configurations to simplify analysis. We show that the reductions are sound, dual of each other and are locally complete. The reductions are also computationally attractive, requiring only local configuration information and modification. These properties establish our reduction calculus as a sound, efficient, and complete theory for scaling up existing analysis techniques.

Disciplines
Computer Sciences

Comments
Wang, A., Talcott, C., Gurney, A., Loo, B., & Scedrov, A., Brief Announcement: A Calculus of Policy-Based Routing Systems, ACM Symposium on Principles of Distributed Computing, July 2012, doi: 10.1145/2332432.2332501

Brief Announcement: A Calculus of Policy-Based Routing Systems

Anduo Wang* Carolyn Talcott† Alexander J. T. Gurney*
Boon Thau Loo* Andre Scedrov*
*University of Pennsylvania, Philadelphia, Pennsylvania, USA †SRI, Menlo Park, California, USA
{anduo,agurney,boonloo}@cis.upenn.edu clt@csl.sri.com
scedrov@math.upenn.edu

ABSTRACT

The BGP (Border Gateway Protocol) is the single inter-domain routing protocol that enables network operators within each autonomous system (AS) to influence routing decisions by independently setting local policies on route filtering and selection. This independence leads to fragile networking and makes analysis of policy configurations very complex. To aid the systematic and efficient study of the policy configuration space, this paper presents a reduction calculus on policy-based routing systems. In the calculus, we provide two types of reduction rules that transform policy configurations by merging duplicate and complementary router configurations to simplify analysis. We show that the reductions are sound, dual of each other and are locally complete. The reductions are also computationally attractive, requiring only local configuration information and modification. These properties establish our reduction calculus as a sound, efficient, and complete theory for scaling up existing analysis techniques.

Categories and Subject Descriptors

C.2.2 [**Computer-Communication Networks**]: Network Protocols

General Terms

Verification, Management

1. INTRODUCTION

The Internet today runs on a complex routing protocol called the *Border Gateway Protocol* or *BGP* [2] for short. BGP enables Internet Service Providers (ISPs) worldwide to exchange reachability information to destinations over the Internet. With policy-based routing, BGP enables network operators to influence routing decisions by independently setting local policies. Each ISP acts as an autonomous system and influences the Internet routing process for its own economic reasons through policy-based routing. This independence leads to fragile networking and makes analysis of policy configurations very complex. Given the set of local policy configurations at each router, a BGP system converges and is said to be safe, if it produces stable routing tables, given any sequence of routing message exchanges. To aid the systematic and efficient study of the policy configuration space, this paper makes the following contributions.

- We propose an abstract model for modeling Internet topology and policies. This abstract model, which we call the *Extended Path Diagram* (EDP), extends prior models [1, 3], and provides a basis for our reduction calculus.
- We present a reduction calculus on policy-based routing systems. In the calculus we provide two types of reduction that transform policy configurations to simplify analysis. Using our EDP model, we show that the reductions are sound, dual of each other and are locally complete. The reductions are also computationally attractive, requiring only local configuration information and modification. As a result, they have the potential to significantly reduce analysis time, and also provide a basis for identifying network instances that have similar configuration patterns.
- Finally, to demonstrate the practical value of our reduction calculus, we have identified several use cases.

2. FORMAL MODEL

The development of our calculus requires support for both specification and analysis, rather than using existing model [1] or analysis structure [3] and switch between them, we develop our own abstract model – *EPD (Extended Path Digraph)*, which combines the strength of both. First, EPD specifies policy configuration by specifying the network topology (which node shares routing information with which) and the routing policy at each node (what are the available paths and routing preference among them). Second, EPD embeds the route preference dependency relation in the network topology, which enables reasoning directly on EPD. The fact that EPD includes the network topology and routing policy makes EPD convenient for reasoning in and about our calculus, especially the reasoning makes use of the network topology change.

We first provide the notations we will use throughout the paper. For a policy configuration, we write V for its network nodes, d for the fixed destination node in V[1], P for the union of all simple paths from nodes in V to d(i.e. P are the available routing paths produced by the routing system under the policy configuration), P_u for the subset of P consisting of paths from u to d, and (u, v) for the one-hop path from node u to v. We use the symbol '$\circ$' for concatenation of paths. Given a path p from v to d, we write $(u, v) \circ p$ or simply $u \circ p$ for the path from u to d. Similarly, given a p from u to v, and a path q from v to node w, we write $p \circ q$ to denote the concatenated path from u to w. Finally for two paths p, q in P_u from the same node u to d, we write $p \prec q$ to denote u's

[1]Routing policies are configured separately for each destination, and assuming the Internet is symmetric, we can focus our discusses on a fixed destination.

PODC'12, July 16–18, 2012, Madeira, Portugal.
ACM 978-1-4503-1450-3/12/07.

preference of p over q[2]. Using these notations, we define the EPD instance associated with a policy configuration as follows:

An EPD instance is a graph $G = \{(V, P, d), E\}$ where (V, P) are the vertexes in the graph, $d \in V$ the particular destination, and E is the set of arcs. V are node vertexes and P are path vertexes, available paths to d. The are two types of arc: (1) transmission arcs, (p, q) where p is proper suffix of q; and (2) preference arcs, (p, q) where p, q are path vertexes in P_u, the paths from u to d, and p is preferred over q according to the route preference relation of u ($p \prec q$).

Closely related with safety is the acyclicity of the configuration. We say a policy configuration is cyclic (acyclic) if its EPD is cyclic (acyclic). An EPD instance $G = \{(V, d, P), E\}$ is cyclic (acyclic) if the arcs E contains at least one (no) cycle. The following characterization was first proved by Sobrinho [3]:

If a policy configuration is acyclic, then it is safe. If a policy configuration is cyclic, then we can construct a trace of routing updates under which the routing system exhibits route oscillation.

3. THE CALCULUS

This section presents a calculus of policy based routing systems, in which one rewrites a configuration by reduction. There are two types of reduction - *duplicate* and *complementary* reductions that only require checking *local* policy configurations at the relevant nodes and their direct neighbors. The basic idea is to incrementally merge two node vertexes into one while preserving safety property. At the top level, the reduction proceeds by repeatedly: (1) locating two reducible nodes; (2) if reducible, merge the two node's local configuration according to the reduction; and (3) rewriting the remainder of the EPD instance to reflect that local change. In the following, assume we are working with a given EPD instance $G = \{(V, d, P), E\}$. We first introduce two auxiliary notions: "consistent node" and "node rewrite" that will simplify our presentation of the calculus.

Two EPD node vertexes u, v in V are consistent, if E does not contain a cycle consisting of only path vertexes in u, v.

Inconsistent vertexes implies the policy configuration is unsafe, and therefore is considered a mis-configuration. Our reductions will only be performed on configurations with no inconsistent pairs. Next, we define "node rewrite".

For two consistent node vertexes u, v in V, u rewrites *to v by follows: Rewrite the path vertex $p \in P_u$ in u to $p' \in P_v$ in v by: If $p = u \circ w \circ r$, and $w \neq v$, then rewrite p to $w \circ r$; If $p = u \circ v \circ r$, then rewrite p to r; For all other cases, abort rewrite. Rewrite the preference among P_u to that among P_v by: Rewrite preference arc (p, q) to (p', q'), where $p(q)$ rewrites to $p'(q')$.*

Duplicate and Complementary Nodes We define two notions of reducible nodes, which we call duplicate and complementary.

For two consistent node vertexes u, v in V, v is a duplicate *of u, if after rewriting v to u, the following conditions hold: (1) v's path vertexes P'_v is equivalent to (or a subset of) the path vertexes P_u; (2) For every preference arc (p, q) in v, there exists (p', q') in u.*

For two consistent node vertexes u, v in V, v is complementary *to u, if for any two path vertexes p in P_u, q in P_v, the following condition holds: For any two node vertexes x, y from u, v's downstream neighbors, which route to the destination through u, v. A preference arc from $x \circ p$ to $x \circ q$ exists in x, iff a preference arc exists from $y \circ p$ to $y \circ q$ in y.*

To merge u, v, if u, v are duplicate, we can simply remove u; if u is complementary to u, then we merge them by: (1) Let path vertexes P_w to be the union of P_u and P_v; (2) Let transmission arcs in w be according to the consensus agreed by u, v's neighbors: transmission arc (p, q) is in w, if $(x \circ p, x \circ q)$ is in x for some downstream neighbor x.

[2] $\prec$ specifies the route preference among paths. It is local to each node, i.e., it only relates paths from same node.

4. PROPERTIES

The key properties of reduction are that the duplicate and complementary reductions preserve safety, that they are dual of each other by nature, and locally complete. Finally, we discuss confluence property of the reductions.

Soundness Our main soundness result is that the reductions rewrite cyclic (acyclic) EPDs into cyclic (acyclic) EPDs. This means that the reductions preserve safety, i.e. we never have false positives or false negatives, with respect to safety property, after applying the reduction.

Duality Duplicate and complementary reductions are *dual*. Assuming EPD G, two node vertexes u, v, the set of their upstream neighbors N_{from} (through which u, v route to the destination) and downstream neighbors N_{to} (which route to the destination through u, v). If all the nodes in N_{from} (N_{to}) can be merged into one node by (multiple steps of) complementary (duplicate) reduction, then u, v are duplicate (complementary) nodes. This duality implies a very interesting practical result: If two nodes' upstream (downstream) are complementary (duplicate), then these two nodes themselves are duplicate (complementary). Moreover, the reduction can be performed in either order: Merge either $N_{from}(N_{to})$ or u, v first.

Local Completeness Consider an EPD G, our reductions are "local reduction" for two nodes u, v in G, in the sense that we only need (and are allowed) to check the configuration at u, v and their direct neighbors (i.e. N_{from} or N_{to}). We assume no knowledge of the rest of configuration. We write N_{rest} to denote these nodes. That is, the local reduction we propose must preserve the safety property for any configuration N_{rest}. Duplicate and complementary reduction are locally complete. We do not exclude the existence of other safety preserving reduction that requires checking policy configuration beyond u, v and their neighbors.

Confluence Duplicate reduction is confluent while complementary is not. There is a counter-example. For a set of nodes V, if they are pair-wise duplicate, that is any pair of u, v in V can be merged by one-step duplicate reduction, then V can be merged into one single node by multiple steps of duplicate reduction, regardless the order in which the nodes merged. A counter-example for complementary nodes, consider an EPD with three node vertexes `u,v,w` who have the same set of downstream neighbors, we construct their path vertexes and preference such that, while `u,v` and `v,w` can be pairwise reduced into `u'` and `w'` respectively. While complementary reduction can be applied to either *u,v* or *v,w*, a further reduction step is not possible.

Acknowledgment

This research is funded in part by NSF grants (CCF-0820208, CNS-0830949, CNS-0845552, CNS-1040672, TC-0905607 and CPS-0932397), AFOSR grant FA9550-08-1-0352, and ONR grant N00014-11-1-0555.

5. REFERENCES

[1] T. G. Griffin and G. Wilfong. An analysis of BGP convergence properties. In *SIGCOMM*, 1999.

[2] Y. Rekhter., T. Li., and S. Hares. A border gateway protocol 4 (bgp-4). RFC 4271, 2006.

[3] J. Sobrinho. Network routing with path vector protocols: theory and applications. In *SIGCOMM*, 2003.