

Discovery of BGP MPLS VPNs

Sarit Mukherjee, Tejas Naik, Sampath Rangarajan

Center for Networking Research
Bell Labs, Holmdel, NJ
sarit@bell-labs.com, {tnaik, sampath}@research.bell-labs.com

Abstract. BGP/MPLS VPN is a mechanism defined in IETF RFC 2547 that allows service providers to use their IP backbone to provide VPN services. This mechanism is based on using BGP to distribute VPN routing information to the routers in the backbone network and using MPLS to forward VPN traffic. MPLS tunnels are created dynamically when needed, which relieves service providers of pre-provisioning tens of thousands of tunnels. BGP/MPLS VPNs allow service providers to define any arbitrary topology with any number of nodes in a VPN. The service provider can create multiple VPNs using the same core network. Currently most of the service providers track 2547 VPNs either manually or by using a provisioning database. Algorithms described in this paper aims at automating this VPN discovery procedure. Using our algorithms service providers can automatically discover VPNs that have already been configured using the current network configuration information.

Keywords: BGP, MPLS, VPN, Discovery, 2547, Route Target, VRF, Topology

1. Introduction

BGP MPLS VPNs as defined in IETF RFC 2547 [1] provide the capability for service providers to use their IP backbone to provide VPN services to their customers. BGP is used to distribute VPN routing information to the routers in the backbone network and MPLS is used to forward VPN traffic. MPLS tunnels are created dynamically when needed. BGP/MPLS VPNs allow service providers to define any arbitrary topology with any number of nodes in a VPN. The service provider can create multiple VPNs using the same core network. Within the context of RFC 2547, a customer site (or more specifically a customer router referred to as a CE router) is connected to the service provider network (or more specifically an edge router on the provider's core network referred to as the PE router) by one or more ports [2][3]. In Figure 1, customer router CE-A is connected to provider edge router PE-A through one port and customer router CE-C is connected to the same provider's edge router PE-A through another port. Thus, multiple CEs could be connected to the same PE as shown in the figure. Within the core network, provider routers (or P routers) function as MPLS Label Switch Routers when forwarding VPN traffic between PE routers.

CE and PE routers exchange routing information using static routing, RIPv2, OSPF or EBGp. A CE router advertises the customer site's local VPN routes to the PE router and learns remote VPN routes from the PE router. After learning local VPN routes

from CE routers, a PE router exchanges this VPN routing information with other PE routers using IBGP. The service provider associates each of the incoming ports at a PE router to a VPN routing and forwarding (VRF) table. This table contains VPN routing information exchanged by the PE router with the CE router connected to that port. In Figure 1, PE-A has two VRFs, VRF-A that contains VPN routing and forwarding information exchanged with CE-A and VRF-C that contains information exchanged with CE-C. A BGP extended community attribute called the Route Target (RT) attribute identifies a collection of VRFs to which a PE router distributes routes. A PE router uses this attribute to export local routes to other VRFs and to constrain the import of remote routes into its own VRFs. For example, in Figure 1 assume that VRF-A exports a RT and VRF-B on PE-B imports this RT. This means, the CE router (CE-B) corresponding to VRF-B knows how to reach hosts behind the CE router (CE-A) corresponding to VRF-A. In order for CE-A to reach hosts behind CE-B, VRF-B needs to export a RT and VRF-A needs to import this RT as well. Once this is done, bi-directional traffic can flow between hosts behind CE-A and hosts behind CE-B. This means, a bi-directional VPN link is established between VRF-A and VRF-B. Thus, the VRFs together with the RTs define the topology of VPNs. In the rest of the paper, when we refer to traffic flow between VRFs we indeed are referring to traffic flow between the CEs connected to the ports on the PE routers on which these VRFs are defined.

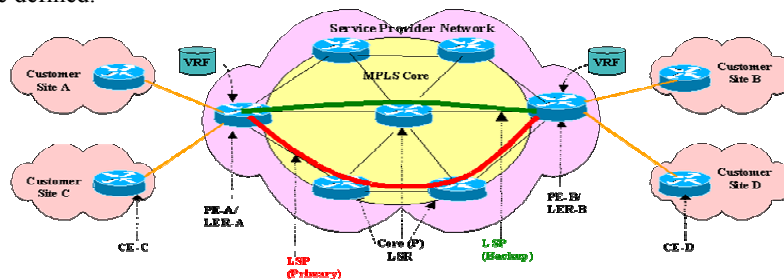


Fig. 1. Example of a BGP MPLS VPN

As illustrated above, a VPN topology can be provisioned using RTs and the export and import of these RTs by the VRFs. Different VPN topologies can be provisioned [2]. Some of the topologies that are normally provisioned include:

- **Single-hub-and-spoke:** In this topology, a single hub VRF can send and receive VPN traffic to a set of spoke VRFs who are not capable of exchanging VPN traffic with each other.
- **Full mesh:** In this topology, a set of VRFs can all exchange VPN traffic with each other. That is, the VRFs are completely connected.
- **Multi-hub-and-spoke:** In this topology, a set of hub VRFs can exchange VPN traffic among each other as well exchange VPN traffic with a set of spoke VRFs. The spoke VRFs cannot exchange VPN traffic with each other.

Figure 2 illustrates these topologies. Figure 2(a) shows a single-hub-and-spoke topology where VRF v_1 is the hub and VRFs v_2, v_3, v_4 and v_5 are spokes. Figure 2(b) is a full-mesh where all VRFs can exchange VPN traffic with each other and Figure 2(c)

is a multi-hub-and-spoke where VRFs v_1, v_2 and v_3 form a full-mesh and a multi-hub and the spokes are v_4, v_5, v_6 and v_7 .

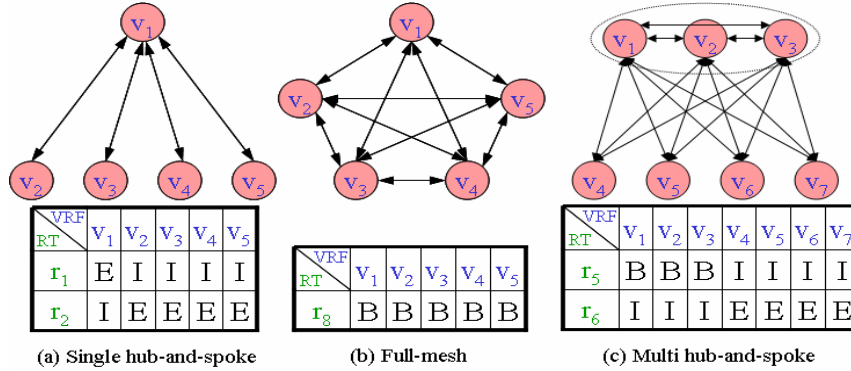


Fig. 2. Different Atomic and Molecular VPN Components

2. Related Work

One important issue that arises after the VPNs are configured is the problem of tracking the configurations of already existing VPNs using the network configuration information. Although tools are available that provide the ability to configure VPNs, locate physical faults that may occur within the VPNs and measure MPLS VPN traffic and quality of service [4][5], to the best of our knowledge, we are not aware of tools that, given the current VRF and RT relationship information, automatically generate the different components that make up the VPN. Such a discovery tool would provide the necessary information to ascertain a) if the VPNs are configured according to specifications, b) find redundancy in the configurations, and c) provide information to visualize the VPN in terms of component topologies described earlier. In this paper, we present an algorithm to implement such a discovery tool. Our algorithm is not specific to any vendor so it can be applied across any vendor. The IETF draft [6] talks about how one PE can discover other PEs in a VPN using a protocol. Our algorithm discovers global view of all the VPNs using offline algorithm. We implemented mandatory steps of algorithm in our company product.

3. Algorithm for Topology Discovery

When the VRFs are provisioned, they may be provisioned using a minimum number of RTs. For example, to provision a full-mesh, only one RT is needed. As long as a single RT is defined on all the VRFs and is exported and imported by all the VRFs, VPN connectivity is established between every pair of VRFs thus leading to a full-mesh topology. Similarly, to provision a single-hub-and-spoke or a multi-hub-and-

spoke only two RTs are needed. One RT will be exported by the (multi) hub which will be imported by all the spokes and all the spokes will import a single RT which will be imported by the (multi) hub. We refer to the largest such components provisioned using the minimum number of RTs as atomic and molecular components as defined below.

Definition 1 [Atomic Component]: The largest single hub-and-spoke with two RTs and the largest full-mesh with one RT are atomic components. Figure 2 (a) and (b) are examples of atomic components.

Definition 2 [Molecular Component]: The largest multi hub-and-spoke with two RTs without any restriction on overlapping links and nodes with atomic components is called a molecular component. Figure 2 (c) is an example of molecular component. Note that it is composed of four atomic components, one full-mesh and three single hub-and-spokes.

In Figure 2, a VRF-RT table is used to represent the export-import relationship between VRFs and RTs. An E entry denotes that the RT represented by the row is being exported by the VRF represented by the column. Similarly, an I entry denotes an import. An entry of B denotes that the RT represented by the row is being both imported and exported by the VRF represented by the column. From the VRF-RT tables above it can be seen that they do represent atomic and molecular components.

An important problem to be solved is the discovery of different components that a VPN is comprised of (that is, the topology of the VPN in terms of different components) and this problem is the focus of this paper. In addition to discovering atomic and molecular components, which are provisioned using minimum number of RTs, we are also interested in discovering basic components such as full-mesh, single-hub-and-spoke and multi-hub-and-spoke even if they are provisioned using more than the minimum number of RTs. In this regard, we define two other types of components.

Definition 3 [Composite Component]: The largest single hub-and-spoke or the largest full-mesh or the largest multi hub-and-spoke components without any restriction on the number of RTs are composite components. Therefore, by definition all atomic and molecular components are composite components.

Definition 4 [Complex VPN]: A complex VPN is a union of composite components.

A composite component is either a single-hub-and-spoke or multi-hub-and-spoke or full-meshes without any regard to the number of RTs that are used. It is clear that each of these components is a union of unique atomic and molecular components (that use the minimum number of RTs). This means, a composite component could be represented as a union of unique atomic and molecular components. By definition, a complex VPN is a union of composite components. Given this definition, it is clear that **a)** the set of atomic components in a complex VPN is unique, and **b)** The set of atomic and molecular components in a complex VPN are unique after the links and nodes belonging to set of atomic components that overlaps with set of molecular components are taken out of the set of atomic components.

The above observation implies that a complex VPN topology can be uniquely represented using a set of atomic and molecular components, whereas representation using composite components may not be unique. The goal of our algorithm is to discover complex VPNs as a composition of atomic and molecular components and in order to do that, we first identify all the atomic components, construct molecular

components from the atomic components, if any and then represent a complex VPN as a union of these atomic and molecular components. The optional goal is to represent a complex VPN as a set of composite components.

3.1 Requirements of the discovery algorithm

In addition to discovering complex VPNs, we pose some additional requirements on our algorithm that optimizes the discovery process as outlined below.

1. Find all the redundant RTs, that is the RTs that are not producing any new topology which is already not discovered. Denote the set of redundant RTs as \mathcal{R} .
2. Find all the unidirectional links. Denote the set of unidirectional links as \mathcal{U} .
3. Find all the atomic and molecular components of the complex VPN. Denote the set of atomic full-mesh components as \mathcal{F} , atomic single hub-and-spoke as \mathcal{S} and molecular multi hub-and-spoke components as \mathcal{M} .
4. Find and express the topology of the complex VPN if it is made of composite components

3.2 The Discovery Algorithm

We assume that route distribution is purely due to BGP MPLS VPN. It is not affected by route redistribution, filter or route maps on PE or CE routers.

Given a description of a VPN (using RTs), it can be decomposed into different sets of components. We use (f_1, f_2, \dots, f_x) to denote a full-mesh created using nodes $f_i, i = 1, \dots, x$. We use $(h \rightarrow s_1, s_2, \dots, s_x)$ to denote a single hub-and-spoke with h as the hub and $s_i, i = 1, \dots, x$ as the spokes. Similarly we use $(h_1, h_2, \dots, h_y \rightarrow s_1, s_2, \dots, s_x)$ to denote a multi hub-and-spoke with $h_i, i = 1, \dots, y$ as the hubs and $s_i, i = 1, \dots, x$ as the spokes. Note that (h_1, h_2, \dots, h_y) is a full-mesh. The steps of our algorithm are enumerated below.

Step 1: Prepare VRF-RT Table: For the given network, let the number of VRFs be n and the number of unique RTs be m . Number the VRFs as v_1, v_2, \dots, v_n and number the RTs as r_1, r_2, \dots, r_m . Prepare a $m \times n$ table, referred to VR , where RT $r_k, 1 \leq k \leq m$ forms the k^{th} row and VRF $v_i, 1 \leq i \leq n$ forms the i^{th} column of the table. Fill in the entries in the VR table with B, E or I according to the specified RTs. The preparation of this table would take $O(nm)$ time.

Note: Remove a row from the VRF-RT table if the row has only one B, all E's or all I's

Above step will remove all RTs that are not capable of forming unidirectional or bidirectional link and hence do not form participate in any VPN. This removes the corresponding RT from the set of RTs.

Step 2: Construct Adjacency Matrix: Construct a graph based on the VRF-RT table as follows [7]. The VRFs are the nodes of the graph. Put a directed edge with label r_k from node v_i to node $v_j, i \neq j$, if RT r_k from VRF-RT table is exported by node v_i and imported by node v_j . Let the edge be represented by $(v_i, v_j)r_k$. Treat B's as both E and I. In an adjacency matrix (AM) representation of the graph, create a $n \times n$ matrix with $AM(v_i, v_j) = r_k$ if there is an RT $r_k, 1 \leq k \leq m$ in the VRF-RT table that is exported by

node v_i and imported by node v_j and $i \neq j$; $i, j = 1, \dots, n$. Note that the diagonal entries of the matrix are left empty. In Step 1, all redundant RTs would have been removed. This means, a directed edge between two VRFs would exist only due to one RT. Even then, in the worst case, to determine the existence of an edge between two VRFs (say v_i and v_j), all the column entries in the VRF-RT table corresponding to VRF v_i need to be checked. This means, to find all edges between all pairs of nodes would take $O(n^2m)$ time.

Step 3: Determine Unidirectional Links: A link qualifies as unidirectional if the nodes it is directed between do not have another link going in the opposite direction. In the adjacency matrix representation, if $AM(v_i, v_j)$ exists, but $AM(v_j, v_i)$ does not, then (v_i, v_j) is an unidirectional link, $i \neq j$; $i, j = 1, \dots, n$. Remove all the unidirectional links from the graph and put them in \mathcal{U} . Therefore,

$$\mathcal{U} = \{ (v_i, v_j)r_k \mid AM(v_i, v_j) = r_k \wedge AM(v_j, v_i) = \Phi, i \neq j, 1 \leq i, j \leq n, 1 \leq k \leq m \}.$$

A simple algorithm to determine this would require all entries in the adjacency matrix to be checked once and thus would take $O(n^2)$ time.

Step 4: Reduce Set of RTs: The general RT reduction algorithm is the following:

1. Denote by binary variable x_{ri} , $1 \leq i \leq m$ if RT r_i is present in the set of minimal RTs.
2. Consider each cell in the adjacency matrix. Let (r_1, r_2, \dots, r_p) represent the set of RTs in that cell. Introduce a constraint as $x_{r_1} + x_{r_2} + \dots + x_{r_p} > 1$.
3. Minimize $\sum_{i=1}^m x_{ri}$ subject to the above set of constraints.
4. Solve the minimization problem and the x_{ri} 's thus achieved give the minimal RT set.

Note that if we want to keep some RT i for some reason then we should make that $x_i = 1$ in the constraint set. If we want to give preference on removal of one RT over another, then the objective function can be changed to

$$\text{Minimize } \sum_{i=1}^m w_i x_{ri},$$

where w_i is a relative weight on the RT. If we want to remove a RT in preference of another, then the former should be given higher weight.

Remove the rows containing the redundant RTs from the VRF-RT Table. Also remove the redundant RT from the cells in the VRF-VRF Table. For the rest of the algorithm, assume that the set of RTs has been reduced following the algorithm. The number of RTs may have been reduced, and the reduced number is still denoted by m , and there is no gap in the sequence of RTs. The algorithm to reduce the set of RTs to a minimum number of RTs reduces to a node covering problem in a graph and is known to be NP-complete. The discovery algorithm does not require this reduction to work correctly and hence this is an optional step.

Step 5: Determine Set of Atomic Full-Mesh Components: If an RT in VRF-RT table has more than one B's then output the set of nodes with B's as a full-mesh. Put them in \mathcal{F} . Now remove the B's from the table. The effect of this is to remove all the corresponding links in the graph, i.e., the entries in AM , but not the nodes. We define $b_k = \{v_i \mid VR(r_k, v_i) = B, 1 \leq i \leq n\}$, which is the set of all the VRFs which both imports and exports r_k (i.e., B in the cell in the VRF-RT table for row r_k). Therefore,

$$\mathcal{F} = \{b_k \mid |b_k| > 1, 1 \leq k \leq m\}.$$

This step requires all entries in the VRF-RT table to be checked once in the worst case and thus would require $O(nm)$ time.

Step 6: Create Set of Candidate Hubs: In order to discover all the atomic single hub-and-spoke components, we start with selecting a hub. A node with out-degree one or more qualifies for this¹. The set is referred to as the set of candidate hubs and is denoted by \mathcal{CH} .

$$\mathcal{CH} = \{v_h \mid \exists i, k, \text{ s.t. } VR(v_h, v_i) = r_k, 1 \leq i \leq n \ 1 \leq k \leq m \}.$$

A simple implementation of this step which consults, in the worst case all entries in the VRF-VRF adjacency matrix, would require $O(n^2)$ time.

Step 7: Create Set of Preferred Hubs: A node in an atomic full-mesh component may become a hub in a molecular multi hub-and-spoke component. It can happen only if the RT used for determining the atomic full-mesh has I in some its entries. In order to facilitate the determination of molecular components, we prepare a set of preferred hubs denoted as \mathcal{PH} . Therefore,

$$\mathcal{PH} = \bigcup_{k=1}^m \{f_k \mid (f_k \in \mathcal{F}) \wedge (\exists i \ VR(r_k, v_i) = I, 1 \leq i \leq n)\}.$$

In the worst case, the set \mathcal{F} may consist of all n VRFs (that is, the complex VPN is made up of one single atomic full-mesh) and this means all entries in each of the columns of the VRF-RT table may have to be checked. This would require $O(nm)$ time.

Step 8: Determine Set of Atomic Single Hub-and-Spokes: We determine how many of hubs from \mathcal{CH} becomes part of an atomic single hub-and-spoke. In order to qualify, there must be two distinct RTs, one where the candidate hub exports to a set of nodes and the other where the candidate hub imports from the same set of nodes.

While there are elements in \mathcal{CH} do the following:

1. For each node $v_h \in \mathcal{CH}$, find all the distinct RTs $r_k, 1 \leq k \leq m$, used to export from v_h . Form the set of spokes $S(v_h, r_k)$ for each distinct RT r_k .

$$S(v_h, r_k) = \{s \mid VR(v_h, s) = r_k\}.$$

2. For each of the set of spokes $S(v_h, r_k), 1 \leq k \leq m$, find the largest subset of nodes that uses the same RT to export to the hub. The cardinality of the largest subset is the in-degree of the hub.

$$In-Degree(v_h) = \max_{1 \leq k \leq m} |\{s \mid VR(s, v_h) = r_j, s \in S(v_h, r_k), r_j \neq r_k, 1 \leq j, k \leq m\}|.$$

3. Find the hub $v_h \in \mathcal{CH}$ with the largest in-degree. If multiple hubs qualify, then select a hub $v_h \in \mathcal{PH}$. Make $(v_h \rightarrow \{s_i \mid VR(s_i, v_h) = r_j, s_i \in S(v_h, r_k), r_j \neq r_k, 1 \leq j, k \leq m\})$. Include this single hub-and-spoke, $(v_h \rightarrow s_1, \dots, s_x)$ in \mathcal{S} . Therefore,

$$\mathcal{S} = \mathcal{S} \cup \{(v_h \rightarrow s_1, \dots, s_x)\}.$$

4. Remove all links associated with this single hub-and-spoke component $(v_h \rightarrow s_1, \dots, s_x)$ from the graph. That is assign $AM(v_h, s_i) = AM(s_i, v_h) = \Phi$.
5. Remove singleton nodes (i.e., nodes with no incoming and outgoing links) from \mathcal{CH} .

¹ Note that since the unidirectional links are removed, in-degree and out-degree of a node are the same.

For each of the VRFs, the (sub) step 1 would take $O(m)$ time. To find the in-degree of each of the hubs in (sub) step 2 would take $O(m)$ time in the worst case considering all nodes that form the spoke for the hub. To find the hub with the largest in-degree in (sub) step 3 would require $O(n)$ time in the worst case. Sub steps 4 and 5 take constant amount of time. Thus, for each of the elements in \mathcal{CH} the time spent in the above steps is $O(m+n)$. The set \mathcal{CH} may contain all n VRFs and thus it would take $O(mn + n^2)$ time to execute the above sub steps.

Step 9: Determine Set of Molecular Multi Hub-and-Spokes: The following steps prepare the set \mathcal{M} .

1. From the set \mathcal{F} , take a new full-mesh component all whose nodes are members of \mathcal{PH} . That is, $b_k \in \mathcal{F} \wedge b_k \subseteq \mathcal{PH}$, $1 \leq k \leq m$. Stop if there are no new full-mesh components for consideration.
2. Check if each of the nodes of the full-mesh component b_k is a hub in the set \mathcal{S} of single hub-and-spoke.
3. Check for each atomic single hub-and-spoke (where the hub $\in b_k$) the RT exported by the hubs is the same and the RT imported is the same one for creating full-mesh b_k . Also check that the RT imported by the hubs is the same.
4. If the test passes, put the full-mesh and the associated single hub-and-spoke components into the multi hub-and-spoke set \mathcal{M} and remove them from \mathcal{F} and \mathcal{S} , respectively. Otherwise go back to step 1.

Each node in set \mathcal{F} could be such that it is not part of a full-mesh component. In this case, for each of these nodes, sub-step would take $O(m)$ time as each RT may have to be checked. Thus, the time required to execute the above sub-steps would be $O(nm)$.

Steps 8 and 9 are the core of the algorithm where the complex VPN is discovered and represented as a composition of atomic and molecular components. This is the main goal of the algorithm. The steps described below are optional and serve to provide more information about the Complex VPN in terms of composite components.

Step 10: Check if the Complex VPN is a Composite Full-Mesh: We determine if the complex VPN is a composite full-mesh. That is every node is directly reachable from every other node. This is verified from the adjacency matrix if each entry in the upper triangular matrix without the diagonal has a valid RT entry in it. If so, the complex VPN is a composite full-mesh. The adjacency matrix can be checked in $O(n^2)$ time.

Step 11: Check if the Complex VPN is a Composite Single Hub-and-Spoke: We determine if the complex VPN is a composite single hub-and-spoke. It is straight forward by making sure that \mathcal{F} and \mathcal{M} are empty and all the single hub-and-spoke components in \mathcal{S} have the same hub. The time for this step in the worst case is $O(n)$ if all VRFs form the hub for some single-hub-and-spoke component.

Step 12: Check if the Complex VPN is a Composite Multi Hub-and-Spoke: We determine if the complex VPN is a composite multi hub-and-spoke by doing the following:

1. Determine the largest full-mesh component in the graph. This is done by finding the largest square sub-matrix with same set of nodes in the rows and columns from the adjacency matrix such that each entry of the sub-matrix has a valid RT in it, except for the diagonal of the sub-matrix which may or not have any entry. Denote

by \mathcal{CF} the set of composite full-mesh which is composed of the nodes of the sub-matrix.

2. From the set \mathcal{S} , combine two single hub-and-spokes into one single hub-and-spoke if they both have the same hub. Continue till no more combinations are possible. The set thus formed is called the composite single hub-and-spoke and represented by \mathcal{CS} .
3. Check if the set of hubs formed from \mathcal{CS} is same as the set \mathcal{CF} . If so check if each single hub-and-spoke component of \mathcal{CS} has the same set of spokes. If the test passes, and \mathcal{CS} contains all the nodes of the network, then the complex VPN is a multi hub-and-spoke topology.

This step requires the largest clique to be identified within a graph [8] as indicated in sub-step 1 and hence is NP-complete [9].

Note that if the RT reduction phase of the algorithm is not conducted, then

- It is possible that after the determination of Set of Molecular Multi Hub-and-Spokes there are still some entries left in the adjacency matrix. The corresponding RTs will be unidirectional and redundant and will not produce any new topology.
- During step 8, remove unidirectional links, if any.

The above algorithm finds all atomic and molecular components and thus the complex VPN that is composed of these components. Ignoring the optional steps (Step 4 and Steps 10, 11 and 12), the complexity of the algorithm is bounded by Step 2 which is the time for constructing the adjacency matrix. Thus the running time of the algorithm is $O(n^2 m)$ where n is the number of VRFs and m is the number of RTs.

We have implemented all the non-optional steps of the algorithm and the algorithm has been used to discover large complex VPNs in terms of atomic and molecular components as part of a Lucent Technologies' network management product. An illustrative example of the steps of the algorithm can be found in the appendix.

4. Open Issues and Future Work

We presented the design of an algorithm for VPN discovery that analyzes complex VPN configurations and represents these VPNs in terms of simple atomic and molecular components. Our algorithm uses a graph model to represent a complex VPN and decomposes this graph to identify the simple components.

The proposed algorithm uses two optional steps, i.e., reduction of set of RTs and determining if the complex VPN is a composite hub-and-spoke, which are NP-complete. For large complex VPNs and open issue is to design efficient approximation algorithms to solve these steps. Another open issue is to extend the algorithm to incrementally discover VPN topology changes. That is, when changes are made in the VRFs of the PE routers, VPN topologies are modified without rerunning the entire algorithm.

In the future, we plan to extend the algorithm to differentiate between intranets and extranets. This involves associating the customer profile information from a provisioning database with the output of our algorithm. In addition we intend to apply the proposed techniques on other types of VPNs that are similar to BGP/MPLS VPNs

such as VPLS. Future work also includes integrating the output of our algorithm to BGP MPLS VPN monitoring systems.

5. References

- [1]E. Rosen, Y. Rekhter, “BGP/MPLS VPNs”, Internet RFC-2547, available at <http://www.ietf.org/rfc.html>, March 1999.
- [2]P. Tomsu, G. Wieser, “MPLS-Based VPNs Designing Advanced Virtual Networks”, Pearson Education, December 2001)
- [3]C. Semeria, “RFC 2547bis: BGP MPLS VPN Fundamentals”, available at http://www.juniper.net/solutions/literature/white_papers/200012.pdf.
- [4]HP Openview Network Services Management Solution for MPLS Networks, Available at <http://www.hp.com>.
- [5]Youngtak Kim, Hyung-Woo Choi, Hyo-Sung Kim, “A QoS-guaranteed DiffServ-aware-MPLS VPN and its Network Management System”, SNPD, 2003
- [6]Hamid Ould-Brahim, Eric C. Rosen, Yakov Rekhter, “Using BGP as an Auto-Discovery Mechanism for Layer-3 and Layer-2 VPNs”, IETF draft, June 2005
- [7]A. Aho, J.Hopcroft, J. Ullman, “The Design and Analysis of Computer Algorithms”, Addison-Wesley Longman Inc., April 1978.
- [8]Harary, “Graph Theory”, Addison Wesley Publishing Company, January 1995.
- [9]Garey and D. S. Johnson. Computers and Intractability, “A Guide to the Theory of NP-Completeness”, Freeman, 1979

Appendix

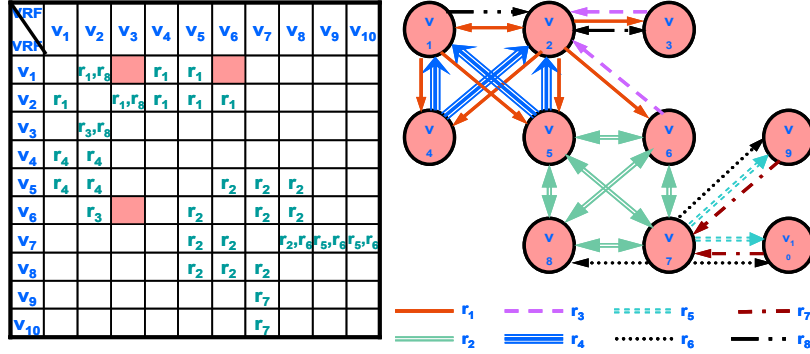
The following example explains important steps of the algorithm to discover all 2547 VPNs using a sample network whose VRF-RT table is shown below. Due to space limitation, we do not elaborate some of the steps.

Step1: Construct VRF-RT Table

VRF RT	v ₁	v ₂	v ₃	v ₄	v ₅	v ₆	v ₇	v ₈	v ₉	v ₁₀
r ₁	B	B	I	I	I	I				
r ₂					B	B	B	B		
r ₃		I	B			E				
r ₄	I	I		E	E					
r ₅							E		I	I
r ₆							B	I	I	I
r ₇							I		E	E
r ₈	E	B	B							

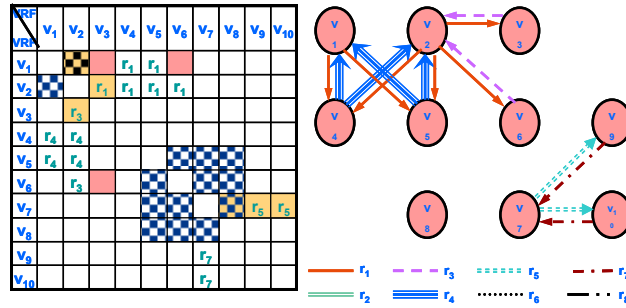
Step 2 and 3: Construct adjacency matrix and remove unidirectional links

$$\mathcal{U} = \{ (v_1, v_3)_{r_1, r_8}, (v_1, v_6)_{r_1}, (v_6, v_3)_{r_3} \}$$



Step 4: Reduce Set of RTs: Minimize $\sum_{i=1}^8 x_i$ subject to $0 \leq x_i \leq 1, 1 \leq i \leq 8$ and $x_1 \geq 1, x_2 \geq 1, x_3 \geq 1, x_4 \geq 1, x_7 \geq 1; x_1 + x_8 \geq 1, x_2 + x_6 \geq 1, x_3 + x_8 \geq 1, x_5 + x_6 \geq 1$. The solution to this problem is $x_1 = x_2 = x_3 = x_4 = x_7 = 1$ and either x_5 or x_6 is 1. We choose $x_5 = 1$. So the reduced set of RTs is $\{r_1, r_2, r_3, r_4, r_5, r_7\}$.

Step 5: Determine Set of Atomic Full-Mesh Components. $\mathcal{F} = \{(v_1, v_2), (v_5, v_6, v_7, v_8)\}$



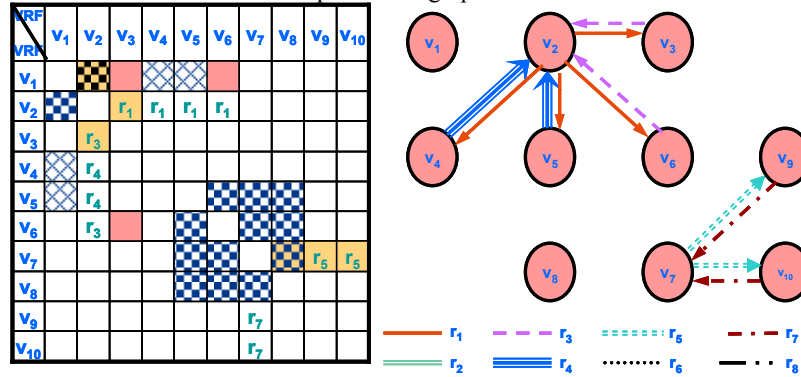
Step 6: Create Set of Candidate Hubs. $\mathcal{CH} = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_9, v_{10}\}$.

Step 7: Create Set of Preferred Hubs. $\mathcal{PH} = \{v_1, v_2\}$.

Step 8: Determine Set of Atomic Single Hub-and-Spokes. For each element of the candidate hub set, we find the set of spokes reachable using one RT. From the set of spokes we then find subsets of nodes that use the same RT to export to the hub. Then we compute the cardinality of the largest such subset.

Spoke set	Elements	Export to hub	In-Degree
$S(v_1, r_1)$	$\{v_4, v_5\}$	$\{v_4, v_5\}$ to v_1 using r_4	$\max \{v_4, v_5\} = 2$
$S(v_2, r_1)$	$\{v_3, v_4, v_5, v_6\}$	$\{v_3, v_6\}$ to v_2 using r_3 and $\{v_4, v_5\}$ using r_4	$\max \{ \{v_3, v_6\} , \{v_4, v_5\} \} = 2$
$S(v_3, r_3)$	$\{v_2\}$	$\{v_2\}$ to using r_1	$\max \{v_2\} = 1$
(v_4, r_4)	$\{v_1, v_2\}$	$\{v_1, v_2\}$ to v_4 using r_1	$\max \{v_1, v_2\} = 2$
$S(v_5, r_4)$	$\{v_1, v_2\}$	$\{v_1, v_2\}$ to v_5 using r_1	$\max \{v_1, v_2\} = 2$
$S(v_6, r_3)$	$\{v_2\}$	$\{v_2\}$ to v_6 using r_1	$\max \{v_2\} = 1$
$S(v_7, r_5)$	$\{v_9, v_{10}\}$	$\{v_9, v_{10}\}$ to v_7 using r_7	$\max \{v_9, v_{10}\} = 2$
$S(v_9, r_7)$	$\{v_7\}$	$\{v_7\}$ to v_9 using r_5	$\max \{v_7\} = 1$
$S(v_{10}, r_7)$	$\{v_7\}$	$\{v_7\}$ to v_{10} using r_5	$\max \{v_7\} = 1$

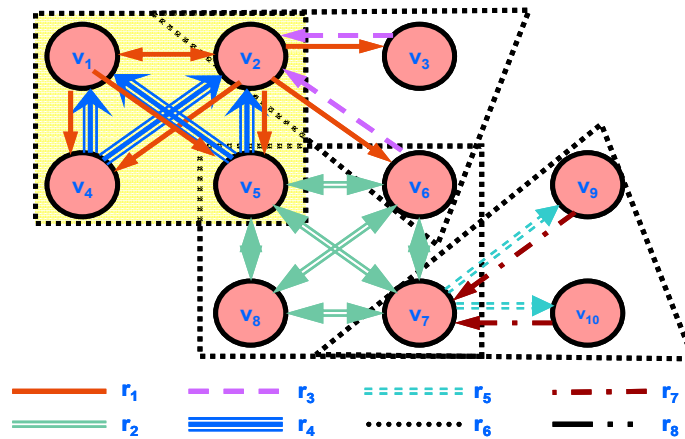
Hubs qualified for selection are v_1, v_2, v_4, v_5, v_7 . We select $(v_1 \rightarrow v_4, v_5)$ since $v_1 \in \mathcal{PH}$. Therefore, $\mathcal{S} = \{(v_1 \rightarrow v_4, v_5)\}$ and now $\mathcal{CH} = \{v_2, v_3, v_4, v_5, v_6, v_7, v_9, v_{10}\}$. After removal of this atomic component the graph is as follows.



Similarly we find atomic components $(v_2 \rightarrow v_3, v_6)$, $(v_2 \rightarrow v_4, v_5)$ and $(v_7 \rightarrow v_9, v_{10})$. Therefore $\mathcal{S} = \{(v_1 \rightarrow v_4, v_5), (v_2 \rightarrow v_3, v_6), (v_2 \rightarrow v_4, v_5), (v_7 \rightarrow v_9, v_{10})\}$ and now $\mathcal{CH} = \{\}$.

Step 9: Determine Set of Molecular Multi Hub-and-Spoke. We see that $(v_1, v_2) \in \mathcal{F}$, and both v_1 and v_2 are hubs in \mathcal{S} . We see that $(v_1 \rightarrow v_4, v_5)$ and $(v_2 \rightarrow v_4, v_5)$ are present in \mathcal{S} . RT exported by v_4 and v_5 is r_4 and it is the same as the one imported by v_1 and v_2 . Therefore,

$$\mathcal{F} = \{v_5, v_6, v_7, v_8\}, \mathcal{S} = \{(v_2 \rightarrow v_3, v_6), (v_7 \rightarrow v_9, v_{10})\}, \mathcal{M} = \{(v_1, v_2 \rightarrow v_4, v_5)\}$$



The complex VPN used in this example is neither a composite full mesh, nor a composite single-hub-and-spoke or a composite multi-hub-and-spoke. Thus, steps 10, 11 and 12 do not apply.