Virtual Network Mapping with Traffic Matrices

Cong Wang and Tilman Wolf
Department of Electrical and Computer Engineering
University of Massachusetts, Amherst, MA 01003
{cwang,wolf}@ecs.umass.edu

ABSTRACT

Network virtualization allows multiple logical networks to coexist on the same physical infrastructure. A key problem that needs to be solved in this context is management of substrate network resources, in particular when mapping virtual network requests to the substrate. Several efficient algorithms have been proposed to provide effective solutions, but they assume that the virtual network request topology is given. In this paper, we show that the use of these topology-based requests may limit the efficiency of physical network usage. As an alternative, we propose a problem formulation and mapping algorithm that is based on traffic matrices to specify virtual network requests.

General Terms

Design, performance

Categories and Subject Descriptors

C.2.6 [**Computer-Communication Networks**]: Internetworking

1. INTRODUCTION

The current Internet architecture is based on an assumption that one single network layer protocol is used among all the devices connected to the network [3]. As the Internet has become more diverse, this principle has become an obstacle for the deployment of innovative new protocols. Network virtualization is one of the key solutions for solving this network ossification problem [1]. Network virtualization allows multiple, independent virtual networks (called "slices") to run on the same physical network (the substrate network) within shared substrate resources. This method is currently used for experimentation in testbeds, but can be deployed at a global level.

An important technical problem for network virtualization is the management of resources in a virtual network infrastructure: When a new virtual network request arrives, how does the infrastructure provider efficiently allocate the physical resources onto that slice? Currently, several efficient mapping algorithms have been proposed to provide solutions for this mapping problem (e.g., D-ViNe [2] and vnm-Flib [4]). These virtual network mapping algorithms use network topologies as the input virtual network request, which means that they assume that each user is able to specify the needed resources in the whole network. In this paper, we reframe the traditional topology-based virtual network mapping problem into a formulation based on *traffic matrices*. This approach reduces constraints and allows for more efficient allocation of resources.

2. VIRTUAL NETWORK MAPPING

Traditional virtual network requests are based on topologies that contain the specific layout pattern of interconnections of the various network elements (e.g., links, nodes) with set of specific description constraints (e.g., CPU capacity, link bandwidth). Figure 1(a) and (c) show a typical examples of the virtual network mapping process. The sample virtual request represented by the network topology shown in Figure 1(a) is mapped to the substrate network shown in Figure 1(c). The requested resources reserved for the virtual network and "subtracted" from the substrate network.

Using a topology-based approach, users of virtual networks need to fully understand the structure of the network in order to provide a valid virtual network request. When lacking knowledge of the substrate network structure, the user topology selection might lead to potential drawbacks that result in low mapping quality or even mapping failure. Specifically, there are two major flaws in current topology-based virtual network mapping:

- **Unnecessary topology constraints:** Nodes and links on the "inside" of the virtual network request determine the topology of the request. However, these nodes and links may not match well with the underlying substrate network. When attempting to map this request, the substrate may need to use excessive resources to accommodate these internal nodes and links.
- **Inconsistent internal topology:** For each node and link of a topology-based request, the bandwidth and computational resources are specified independently. It is possible that a request exhibits internal inconsistencies where resource requests do not match with feasible traffic patterns. An example is shown in Figure 1(a), where node d requests 30 units of bandwidth to c, but only can receive 20 units total from node a and b.

Based on the observations, we believe that avoiding explicit specification of internal topology constraints can improve virtual network mapping. We propose the use of a *traffic matrix* as input for virtual network mapping. Instead of specifically describing each node and link condition of the

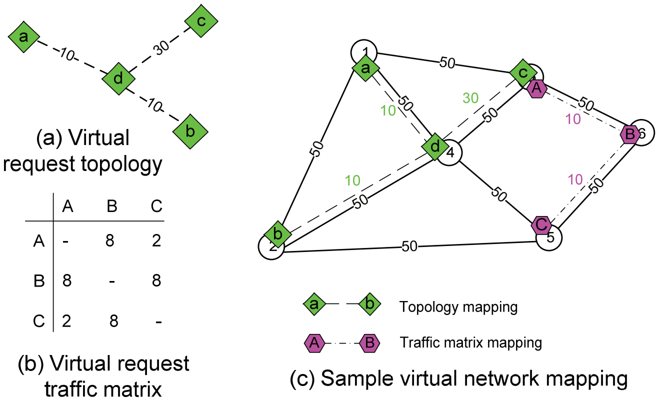

Figure 1: Topology-based and traffic-matrix-based virtual network mapping.

whole network, the traffic matrix only specify the necessary end nodes, the traffic that flows between each pair of the end nodes, and node location constraints. The internal topology of a virtual network then is determined by the mapping onto the substrate. Traffic-matrix-based virtual network mapping is illustrated in Figure 1(b). Although the traditional topology-based request and the traffic-matrix-based request represent the same amount of network end-to-end traffic, the lacking constraints in the latter make it possible for the mapping algorithm to "compress" the virtual network request within the substrate. Thus, traffic-matrix-based mapping requires less substrate resources.

3. EVALUATION

To demonstrate the effectiveness of traffic-matrix-based mapping, we implemented a single stage virtual network mapping formulation $VHub$, that could simultaneously map traffic matrices onto the physical network. Our VHub technique formulates the virtual network mapping process as a mixed integer program that is based on the p-hub median problem. For comparison, we also implemented two other mapping algorithms (D-ViNe and vnmFlib) that represent traditional topology-based virtual network mapping.

For the experimental evaluation, we used requests with five end-nodes. No internal topology was specified for the traffic matrix based request; and for topology-based requests, we generate all possible topologies with three internal nodes and link bandwidth requests that are consistent with shortest path routing of the traffic specified in the traffic matrix. For each of the request, we successively map the same request onto the substrate until the first mapping failure occurs. Note that for each of the topology based mappings, the internal network structure is fixed, however, since no internal topology specified for traffic matrix approach, this formulation is able to find any internal nodes and links that are most beneficial for mapping.

Figure 2 shows the number of virtual networks that can be accommodated and the processing time for all variations of the topologies based requests and for the unique traffic matrix. The vnmFlib algorithm successfully mapped 47 virtual requests, D-ViNe successfully mapped 26 virtual requests, and traffic-matrix-based VHub successfully mapped 77 vir-

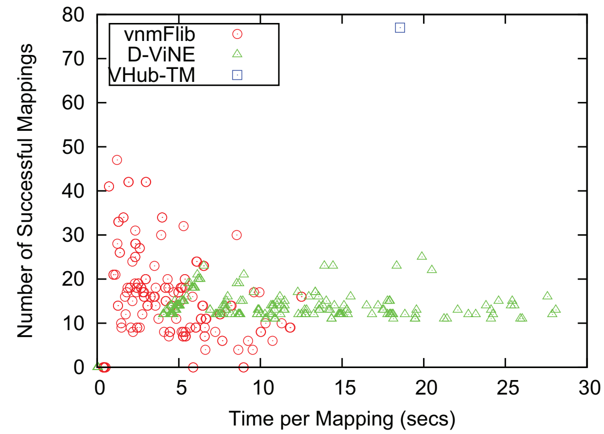

Figure 2: Number of successful virtual network mapping vs. computation time for different algorithms.

tual request (which is 63% more than vnmFlib and 196% more than D-ViNe). Since there is no single topology that provides an ideal basis for mapping under all conditions, traffic-matrix-based mapping clearly outperforms topology-based mapping. The computation time for traffic-matrix-based VHub is slightly longer than the average run time for topology based mapping, but the significant improvement in resource utilization can easily justify this additional cost.

4. CONCLUSION

The mapping of virtual network requests to the substrate is an important operational aspect of network virtualization. Choosing a suitable input for virtual network requests is important. In this paper, we provided a reformulation of the virtual network mapping problem based on traffic matrices rather than internal topologies. Our results show that this approach improve the effectiveness of virtual network mapping. We believe that this change in problem formulation is important step in enabling efficient operation of current and future virtualized networks.

5. REFERENCES

[1] Anderson, T., Peterson, L., Shenker, S., and Turner, J. Overcoming the Internet impasse through virtualization. *Computer 38*, 4 (Apr. 2005), 34–41.

[2] Chowdhury, N., Rahman, M., and Boutaba, R. Virtual network embedding with coordinated node and link mapping. In *IEEE INFOCOM* (2009), pp. 5634–5640.

[3] Clark, D. D. The design philosophy of the DARPA Internet protocols. In *Proc. of ACM SIGCOMM 88* (Stanford, CA, Aug. 1988), pp. 106–114.

[4] Lischka, J., and Karl, H. A virtual network mapping algorithm based on subgraph isomorphism detection. In *Proceedings of the 1st ACM workshop on Virtualized infrastructure systems and architectures* (Aug 2009), ACM New York, NY, USA, pp. 81–88.