**Computer Network Project Proposal- Virtual Network Embedding Algorithms**

**Introduction**

Network virtualization is recognized as an enabling technology for the future Internet. It aims to overcome the resistance of the current Internet to architectural change. It is introduced as a means to evaluate new protocols and services, it has already been actively used in research testbeds like G-lab or 4WARD, applied in distributed cloud computing environments and is, by now, seen as a tool to overcome the resistance of the current Internet architecture. Application of this technology relies on algorithms that can instantiate virtualized networks on a substrate infrastructure, optimizing the layout for service-relevant metrics. This class of algorithms is commonly known as “Virtual Network Embedding (VNE)” algorithms. In my project, my work is to do a survey of current research in the VNE algorithms and implement and optimize two of the algorithms into C++ code. The Algorithms are called BFSN VNE Algorithm and BFSN-HEM VNE Algorithm based on the Breadth First Search Tree and Heavy Edge Matching (HEM) technique and embed coarsened virtual networks on best-fit sub-substrate networks.

The remainder of this report is organized as follows: Section II we discuss the related work on VNE problem and our work to this problem including the shortcomings and our innovation. Section 3 presents the VN embedding model, problem formulation and the proposed algorithms. Section 4 describes the C++ code and the program running result and analysis.

**Background and related work**

Many researches have been done for efficient VN embedding. Zhu and Ammar (Zhu & Ammar, 2006) proposed two VN embedding algorithms. The first algorithm is static VNE algorithm, where allocated substrate resources are fixed throughout the VN lifetime. Heuristics and adaptive optimization strategies are used to improve the performance of the proposed algorithm. The second algorithm reconfigures the embedded VNs to increase the utilization of the underlying substrate resources. However, the proposed algorithms deal only with offline embedding problem (all VNRs are previously known). On the other hand, in cloud computing data centers, VNE problem is online problem, where new VNRs arrive over time.

Some of existing works proposed VN embedding algorithms to embed VNRs in distributed cloud computing environments (Samuel et al, 2013; Houidi et al, 2008a; Xin et al, 2011; Lv et al, 2010). Houidi et al. (Houidi et al, 2011) proposed exact and heuristics VN embedding algorithms, which split virtual network requests using max-flow min-cut algorithms and linear programming techniques. Leivadeas et al. (Leivadeas et al, 2013) proposed VN embedding algorithm based on linear programming. The proposed algorithm partitions VNRs using partitioning approach based on Iterated Local Search. Houidi et al. (Houidi et al, 2008b) proposed distributed VN embedding algorithm, which is performed by agent-based substrate nodes. The authors proposed VN embedding protocol to allow communication between the agent-based substrate nodes. However, the proposed algorithm assumes that all VNRs are previously known.

**Problem Formulation**

**Substrate network (SN):** We model the substrate network as a weighted undirected graph = ( , ), where is the set of substrate nodes and is the set of substrate links. Each substrate node ∈ is weighted by the CPU capacity, and each substrate link ∈ is weighted by the bandwidth capacity.

**Virtual network (VN):** virtual network is modeled as a weighted undirected graph=(,) , where is the set of virtual nodes and is the set of virtual links. Virtual nodes and virtual links are weighted by the required CPU and bandwidth, respectively.

**Virtual network requests (VNR):** the VN request in the set of all VN requests is modeled as (,,) where is the required VN to be embedded, is the arrival time, andis the lifetime. When arrives, substrate nodes’ CPU and substrate links’ bandwidth are allocated to achieve the . If the substrate network does not have enough resources to achieve , is rejected. At the end of lifetime, all allocated resources to are released.

**Virtual Network Embedding (VNE):** embedding on SN is defined as a map M :→(,), where ⊆, and ⊆ , where is the set of all loop free substrate paths in . Embedding can be decomposed into node and link mapping as follows:

Nodemapping: : →

Linkmapping: : : →

**Virtual Network Embedding Revenue:** the revenue of embedding the revenue of embedding at time is defined as the sum of all required substrate CPU and substrate bandwidth by at time .

at time is defined as the sum of all required substrate CPU and substrate bandwidth by at time .