Conflict-Based Search for the Virtual Network Embedding Problem

Yi Zheng, Srivatsan Ravi, Erik Kline, Sven Koenig, T. K. Satish Kumar
yzheng63@usc.edu, sravi@isi.edu, kline@isi.edu, skoenig@usc.edu, tkskwork@gmail.com

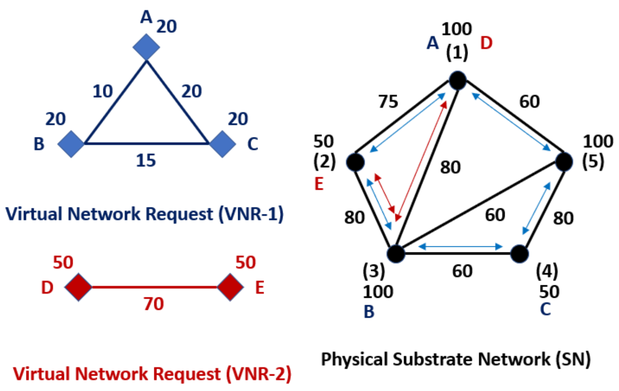
USC University of Southern California

Abstract

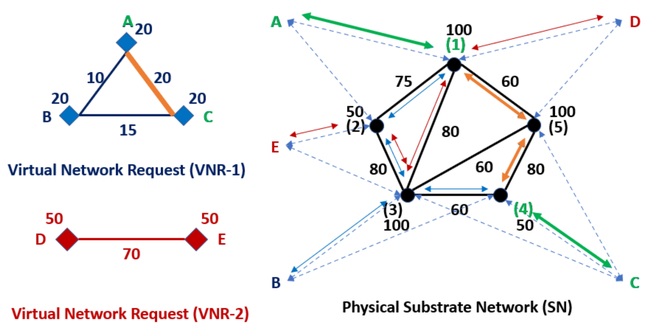
In emerging network virtualization architectures, service providers will be able to create many heterogeneous virtual networks and offer customized end-to-end services by leasing shared resources from infrastructure providers. The Virtual Network Embedding (VNE) problem is central to such technology. It involves the proper allocation of CPU and bandwidth resources available in a physical substrate network to meet the demands of multiple virtual networks. Combinatorially, the VNE problem is a problem in resource management that is NP-hard to solve. In this paper, we present a novel version of the Conflict-Based Search (CBS) algorithm for solving the VNE problem. Our approach, called VNE-CBS, is inspired by the success of the CBS framework in the Multi-Agent Path Finding domain. We successfully address the unique challenges in applying the CBS framework to the VNE problem, and, in doing so, we pave the way for overcoming a crucial issue in Internet ossification via heuristic search methods. On the theoretical front, we show that, unlike many existing algorithms, our algorithm is complete and optimal. On the experimental front, we show that our algorithm significantly outperforms other state-of-the-art methods on various benchmark instances for both the offline and online versions of the VNE problem.

1 Virtual Network Embedding

An example of the VNE problem:

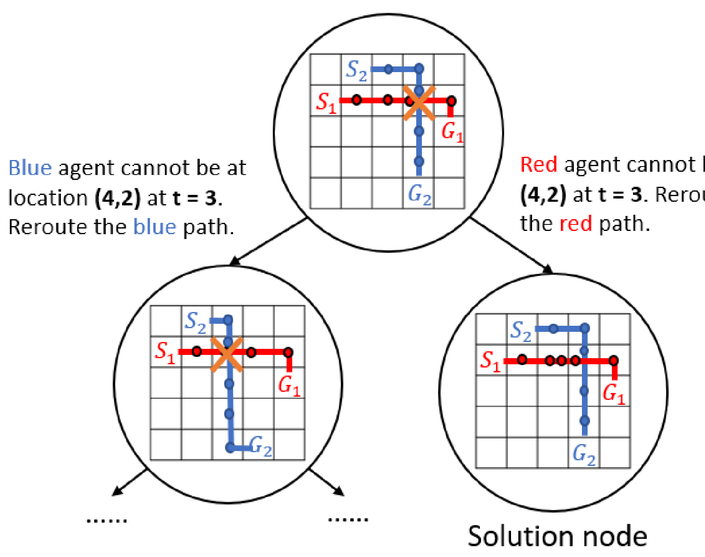


The cost of a VNE mapping refers to the CPU and the bandwidth capacities spent by the SN to map a VNR.
Solve the VNE problem as a multi-path coordination problem:



For example, path A-(1)-(5)-(4)-C represents the node mappings of VNR vertices A and C and the edge mapping of VNR edge A-C.

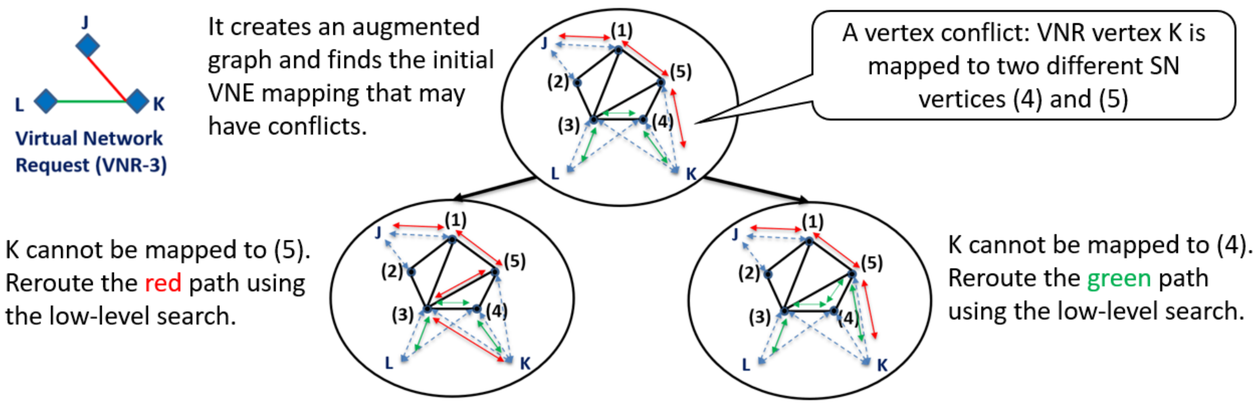
2 Conflict-Based Search



The CBS algorithm is a popular algorithm in the multi-agent pathfinding (MAPF) domain. The high-level search of CBS resolves conflicts between paths, and the low-level search (e.g., A*) finds a path for an agent. CBS keeps expanding and resolving the conflicts until it finds a set of conflict-free paths. The cost of each node is the sum of the travel cost. The high-level search is a best-first search that expands nodes with the smallest cost first.

3 VNE-CBS

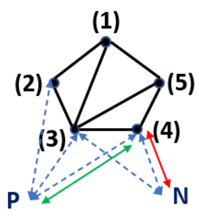
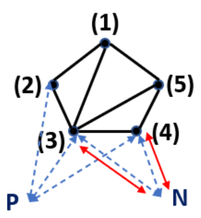
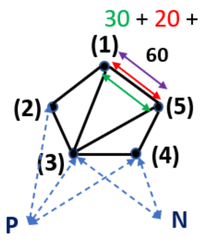
We adapt the CBS framework to the VNE problem:



It keeps expanding nodes and resolving conflicts until a valid VNE mapping is found.

- The cost of each node is the cost of its VNE mapping.
- When two nodes have the same VNE mapping cost, it chooses the node with the fewer number of conflicts.
- Bounded-suboptimal version of VNE-CBS: the high-level search is a focal search with a suboptimality factor $w \geq 1$.
- The low-level search uses a tie-breaking rule to reduce the potential number of vertex conflicts that the returned path may cause in the VNE mapping. It also only maps a VNR vertex to an SN vertex that has enough CPU capacity.

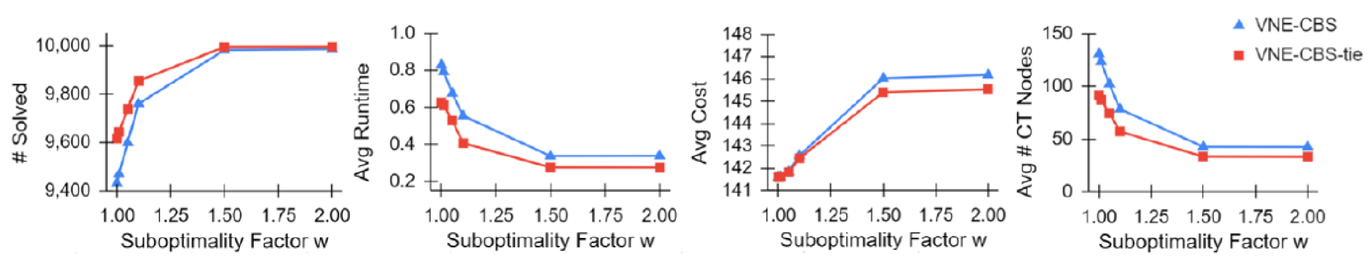
We also define the conflicts for the VNE problem:

	Vertex Conflicts		Bandwidth Capacity Conflict
Example			
Definition	Two VNR vertices (from the same VNR) are mapped to the same SN vertex.	A VNR vertex is mapped to two different SN vertices.	A VNR edge is mapped to an SN path with an SN edge that does not have sufficient bandwidth capacity.
How to Resolve	Create two child nodes: stop P from being mapped to (4); stop N from being mapped to (4).	Create two child nodes: stop N from being mapped to (3); stop N from being mapped to (4).	Create three child nodes: stop the green/red/purple path from using the SN edge (1)-(5).

4 Experimental Results

Algorithm	# Solved	# Timeout	# No Solution	Avg Runtime (s)	Avg Cost	Avg # CT Nodes
D-ViNE	10,000	0	-	1.885	223.935	-
R-ViNE	10,000	0	-	1.904	224.765	-
G-SP	10,000	0	-	0.022	273.085	-
G-MCF	10,000	0	-	0.495	234.178	-
VNE-CBS-w1.00	9,432	568	0	0.829	141.613	130.899
VNE-CBS-w1.00-tie	9,616	384	0	0.627	141.613	91.400

Table 1: Offline Setting: number of VNR vertices $\in_u [2, 10]$; VNR vertex CPU requirement $\in_u [0, 20]$; VNR edge bandwidth requirement $\in_u [0, 50]$.



This work at the University of Southern California is supported by DARPA under grant number HR001120C0157 and by NSF under grant numbers 1409987, 1724392, 1817189, 1837779, 1935712, and 2112533.