Advance Reservation for Virtual Overlay Network Services

Feng Gu[1], Chongyang Xie[1], Min Peng[2], Çiçek Çavdar[3], Samee Khan[4], Nasir Ghani[1]
[1]University of New Mexico, [2]Wuhan University, [3]KTH Royal Institute of Technology,
[4]North Dakota State University

ABSTRACT
Network advance reservation is an area of growing interest and focus today. However most proposed solutions in this field have only focused on the scheduling of point-to-point client connections. Now as applications and user communities continue to expand, there is a growing need to extend advance reservation support for multipoint services. To address this concern, a heuristic solution is presented for virtual overlay network scheduling. The scheme uses graph-theoretic schemes and is analyzed in detail using network simulation.

Keywords: Advance reservation, network scheduling, topology overlay, virtual network services.

1. INTRODUCTION

Advance reservation (AR) services have seen increased focus and interest in recent years. In particular, the ability to schedule future connection requests is becoming an integral requirement for a range of applications in grid-computing, e-science, storage backup, special event broadcasting, etc. As a result, a wide range of AR scheduling algorithms have been studied in recent years [1]-[6], including optimization formulations and graph-theoretic heuristics. Others have also addressed broader topics such as AR survivability [7], re-routing schemes [8], and even distributed solutions [9].

Nevertheless, with expanding applications there is a growing need to extend AR capabilities to support *multipoint* service connectivity between dispersed users/ sites. For example, ubiquitous *virtual private network* (VPN) and/or overlay network services have been widely used to connect organizations and communities of interest over common physical network infrastructures [10]-[15]. However related studies in VPN design have only looked at a-priori or "on-demand" provisioning models for *immediate reservation* (IR). As a result there is a critical need to address future scheduling of multipoint services as well, termed herein as *virtual overlay network scheduling* (VONS). To the best of our knowledge, there are no known studies that address this particular area, i.e., combining AR scheduling with VPN overlay provisioning. Hence this forms the key motivation for the work herein.

Overall, this paper is organized as follows. Section II first presents a brief overview of the existing work in AR provisioning and overlay services. Next, Section III presents a heuristic solution for the VONS problem using graph-theoretic schemes. Section IV then presents detailed performance analysis results for the scheme using network simulation. Finally conclusions and directions for future work are presented in Section V.

2. BACKGROUND

Researchers have studied many different AR schemes for bandwidth-provisioning networks. A broad and in-depth introduction to the area is presented in [1], including detailed discussions on various possible AR service models. Meanwhile [2] outlines scheduling schemes for users with variable starting time/fixed duration requests using an augmented Bellman-Ford approach. The goal here is to accommodate incoming requests at the earliest possible time. [3] extends this work to handle request variations with different path and bandwidth constraints, e.g., variable/fixed path, variable/fixed bandwidth. Detailed polynomial and pseudo-polynomial complexity scheduling algorithms are proposed here, but akin to [2], detailed simulation analysis studies are not done. Meanwhile [4] looks at two different AR schemes for scheduling fixed path/variable bandwidth and variable path/variable bandwidth AR requests. Both of these problems are shown to be NP-complete and related heuristics presented for each. Overall results show good performance gains over greedy algorithms here.

Others have also looked at AR scheduling for optical *dense wavelength division multiplexing* (DWDM) networks. For example, [5] studies *routing and wavelength assignment* (RWA) in the AR context and develops three lightpath scheduling algorithms. However performance analysis results are not presented here. Meanwhile [6] studies lightpath RWA scheduling in "all-optical" networks and compares several spatio-temporal optimizations against sequential RWA schemes (yielding good improvements in setup success rates). Finally, some have also looked at broader aspects of the AR problem, including protection [7], re-routing [8], and distributed implementations [9].

Meanwhile virtual overlay networking design has also seen much focus over the years, with researchers detailing a host of solutions based upon optimization and heuristic approaches. Notable examples include the *resilient overlay network* (RON) study [10] which develops a static overlay for best-effort Internet networks to improve resiliency. The *service overlay network* (SON) study [11] also addresses *quality of service* (QoS) support using queuing theory and optimization methods to implement static partitioning and oversubscription. Meanwhile "dynamic" topology overlays have also been considered. For example, [12] proposes a *virtual network* (VN) scheme for node/link selection with reconfiguration, whereas [13],[14] study multipoint Ethernet

978-1-4577-0882-4/11/$26.00 ©2011 IEEE

overlays in over next-generation SONET/SDH networks. Finally, numerous efforts have addressed *virtual topology* (VT) design for optical DWDM networks, see survey in [15]. Most of these schemes have pursued optimization-based formulations for a-priori virtual link sets.

In summary, existing AR studies have only looked a *point-to-point* connection scheduling. Meanwhile VPN and overlay network provisioning schemes have only treated immediate user demands. Indeed, no efforts have looked at future scheduling of multipoint "network overlay" type requests. Hence there is a strong need to investigate new schemes for *virtual overlay network scheduling* (VONS), i.e., combining the fields of AR and VPN/overlay provisioning. These issues are now investigated further.

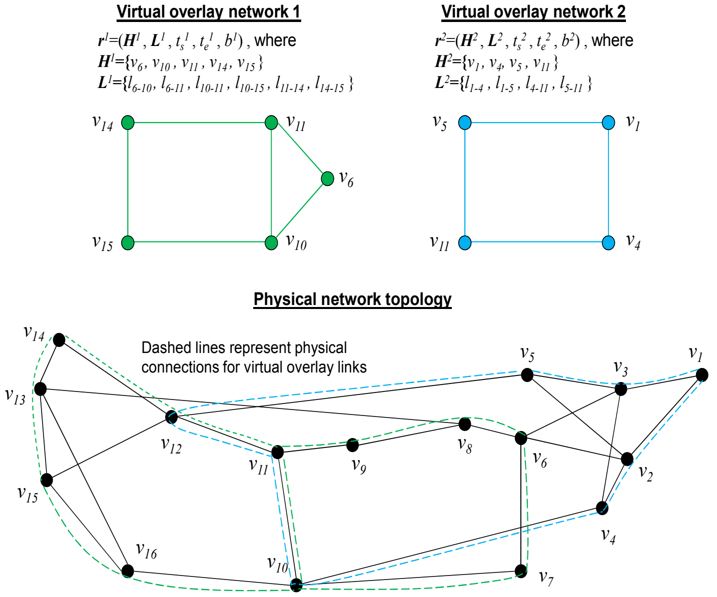
Virtual overlay network 1
r^1=(H^1, L^1, t_s^1, t_e^1, b^1), where
H^1={v_6, v_{10}, v_{11}, v_{14}, v_{15}}
L^1={l_{6-10}, l_{6-11}, l_{10-11}, l_{10-15}, l_{11-14}, l_{14-15}}
v_{14}
v_{11}
v_6
v_{15}
v_{10}
Virtual overlay network 2
r^2=(H^2, L^2, t_s^2, t_e^2, b^2), where
H^2={v_1, v_4, v_5, v_{11}}
L^2={l_{1-4}, l_{1-5}, l_{4-11}, l_{5-11}}
v_5
v_1
v_{11}
v_4
Physical network topology
Dashed lines represent physical connections for virtual overlay links
v_{14}
v_{13}
v_{12}
v_{15}
v_{16}
v_{11}
v_{10}
v_9
v_8
v_5
v_3
v_1
v_6
v_2
v_4
v_7

Figure 1. Virtual overlay network services.

3. VONS HEURISTIC SOLUTION

A heuristic VONS solution is now presented for regular bandwidth-provisioning networks (with further considerations for optical DWDM settings left for future study). The proposed solution uses graph-theoretic algorithms and assumes that all overlay requests arrive in a random "on-demand" manner, i.e., reflective of real-world settings as opposed to idealized a-priori settings. However, before presenting this approach, the requisite notation is first introduced (all vector and set entities denoted in bold).

The overall physical network is modeled as a graph topology, $\boldsymbol{G}(\boldsymbol{V},\boldsymbol{E})$, where $\boldsymbol{V} = \{v_1, v_2, \ldots\}$ is the set of network router/switch nodes and $\boldsymbol{E} = \{e_{ij}\}$ the set of network links, i.e., e_{ij} is the link between v_i and v_j. Furthermore, without loss of generality, all links have fixed capacity, C, and connectivity is assumed to be bidirectional, i.e., two opposing uni-directional links between neighboring nodes. In addition, it is assumed that each link e_{ij} has an associated bandwidth-time function, i.e., $c_{ij}(t)$, which tracks the used capacity at future time instants. Now the n-th overlay request is denoted by the 5-tuple $\boldsymbol{r}^n = (\boldsymbol{S}^n, \boldsymbol{L}^n, t_s^n, t_e^n, b^n)$, where $\boldsymbol{S}^n$ is the set of nodes/sites ($\boldsymbol{S}^n \in \boldsymbol{V}$), $\boldsymbol{L}^n$ is the set of desired overlay links between node sites in $\boldsymbol{S}^n$, t_s^n is the start time, t_e^n is the stop time, and b^n is the requested bandwidth, $b^n \leq C$. Hence, the client is basically requesting setup of a virtual network to interconnect its sites in $\boldsymbol{S}^n$ via a set of "virtual" overlay links given in $\boldsymbol{L}^n$ (i.e., implemented using physical network connections, see Fig. 1). Note that this AR service model only assumes fixed request intervals, and future variants can relax this assumption.

The complete pseudocode for the VONS heuristic is shown in Fig. 2 and assumes random

```
/* Given incoming request $r^n$=($S^n$,$L^n$,$t_s^n$,$t_e^n$,$b^n$) */
Generate temporary graph copy $G'(V,E)$=$G(V,E)$
Remove non-feasible links in $G'(V,E)$, i.e., $c_{ij}(t)<b^n$ in [$t_s^n$,$t_e^n$]

/* Loop and provision all overlay links in request */
for $i$ = 1 to |$L^n$|
    if (minimum hop count)
        - Assign unity weight to links in $G'(V,E)$
        - Compute Dijkstra's shortest-path for $i$-th overlay link
    else if (minimum distance)
        - Assign dynamic weight to links in $G'(V,E)$, Eq. (1)
        - Compute Dijkstra's shortest-path for $i$-th overlay link
    else (maximum bottleneck)
        - Compute k-SP between end-points of $i$-th overlay link
        - Select path with maximum bottleneck bandwidth
end

if (all overlay link connections in $L^n$ routed)
    Setup successful, copy $G'(V,E)$ → $G(V,E)$
else
    Reject request
```

Figure 2. VONS heuristic algorithm.

"on-demand" request arrivals. The algorithm basically loops through all requested overlay links in the overlay topology and attempts to route each using a given *traffic engineering* (TE) strategy. Note that all of these computations are done over a temporary copy of the network graph, ***G'*(*V*,*E*)**, derived from ***G*(*V*,*E*)** by removing *non-feasible* physical links, i.e., those without sufficient capacity in the request interval, $c_{ij}(t)<b^n$ in $[t_s^n,t_e^n]$. Furthermore, three different TE routing approaches are considered here for AR, i.e., *minimum hop count*, *maximum bottleneck capacity*, and *minimum distance*. These strategies are now detailed further:

Minimum hop count: This scheme assigns unity weights to all links and finds the shortest feasible route in ***G'*(*V*,*E*)**, i.e., via Dijkstra's algorithm. This approach pursues pure resource minimization (as it does not account for link loads) and may cause saturation on certain network links.

Maximum bottleneck: This scheme ignores path lengths and instead chooses an end-to-end route with the maximum available capacity. Specifically, the *k-shortest paths* (k-SP) are computed between the source and destination nodes, and the one with the maximum residual capacity (on its bottleneck link during the request interval) is selected. This is deemed as a pure load-balancing approach and can give longer path lengths. Note that *k*-SP computation generally entails much higher complexity.

Minimum distance: This scheme achieves a balance between the above two strategies by assigning dynamic weights to all links. Namely, the cost of each link is set as inversely-proportional to the lowest residual capacity in the request interval, i.e., cost for link e_{ij} is given as:

$$\omega_{ij} = C / (\chi + \varepsilon), \tag{1}$$

where

$$\chi = \min \{b_{ij}(t)\} \text{ in } [t_s^n, t_e^n] \tag{2}$$

and ε is a small value chosen to avoid division errors.

The overall algorithm in Fig. 2 only returns success if all overlay link connections are successfully routed over the temporary graph. In this case the network resource graph is also updated, i.e., ***G'*(*V*,*E*)→ *G*(*V*,*E*)**.

4. PERFORMANCE ANALYSIS

The performance of the heuristic VONS solution is analyzed using specialized simulation models developed in *OPNET Modeler*^{TM}. These tests consider two topologies, including the ubiquitous NSFNET with a moderate node degree of 3.12 (16 nodes, 25 links) and the denser *Deutsche Telekom* (DT) backbone with a higher node degree of 3.85 (27 nodes, 52 links). All network nodes are generic IP *multi-protocol label switching* (MPLS) routers with 10 Gbps links. Now the client overlay requests have random exponentially-distributed holding and inter-arrival times, with means μ and λ, respectively. In particular, a scaled mean holding time of μ= 600 s is used and the mean inter-arrival times are adjusted according to load. Carefully note that this mean holding time is just a relative figure and does not necessarily reflect real-world values.

Meanwhile all overlay topology sizes are uniformly varied from 4 – 6 nodes with corresponding (overlay link) bandwidth requests ranging uniformly from 200 Mb/s to 1.0 Gb/s in 200 Mbps increments, i.e., to model fractional Ethernet demands. Here the node/sites in the overlay topologies are randomly selected from the network nodes. Furthermore the virtual overlay links are selected using the random Inet topology generator [16] to achieve a node degree of about 2.5. All runs are averaged over 500,000 requests and a modified Erlang loading metric is used to account for connection group sizes, as in [14]:

$$\text{Modified Erlang load } = \sum_{n=4}^{6} (n-1) \cdot \frac{\mu}{\lambda}, \tag{3}$$

where the overlay topology sizes range from 4-6 nodes and the 1/λ represents the mean inter-arrival rate.

Initial tests are done to measure overlay request blocking rates for the various TE (overlay link) routing strategies, Fig. 3. These results show that the minimum distance heuristic (Eq. 1) consistently gives the lowest blocking for both topologies. In particular this scheme yields less than half the blocking rate of the maximum bottleneck scheme at low-medium input loads (as shown on log scale in Fig. 3). By contrast the maximum bottleneck scheme only yields marginal (negligible) blocking reduction over the minimum hop count scheme, despite its sizeable increase in complexity, i.e., *k*-SP computations versus Dijkstra's shortest path.

Next, the average path lengths of the computed routes for the overlay links (in the overlay topologies) are plotted in Fig. 4. As expected, the minimum hop count routing scheme gives the lowest utilization of all. Nevertheless, the minimum distance heuristic provides very competitive performance here, with average path length values coming to within 2% of the minimum hop count scheme. Conversely, the hop count utilization for the maximum bottleneck scheme is notably higher, by about 15% for both topologies.

Carefully note that the VONS heuristic in Fig. 2 tries to setup all overlay link (connections) in the order in which they are listed in the request, i.e., random. Hence modified versions of the heuristic were also tested to better "shuffle" the sequence in which the overlay links are attempted, e.g., longest-path first, etc. However, these renditions showed no noticeable reduction in blocking over the base random ordering scheme. Overall, these findings indicate that the minimum distance scheme gives the best performance tradeoff in terms of minimizing blocking and lowering resource consumption on overlay link routes.

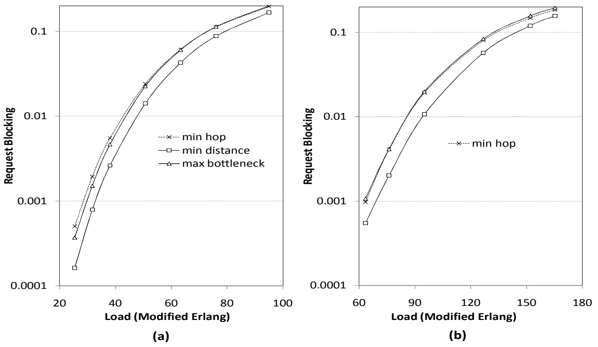

Figure 3. Overlay request blocking:
a) NSFNET, b) DT.

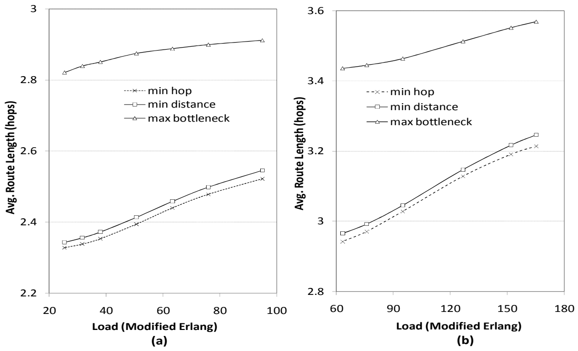

Figure 4. Average path lengths:
a) NSFNET, b) DT.

5. CONCLUSIONS

There is a growing need to develop advance reservation scheduling algorithms for overlay network services. This paper addresses this concern and proposes a graph-theoretic heuristic solution to schedule random "on-demand" virtual overlay topology requests. The scheme incorporates a range of traffic engineering objectives, including resource minimization and load-balancing. Detailed simulation results show that dynamic link-load weighting gives the best performance/tradeoffs in terms of blocking reduction and resource utilization. Future efforts will look at developing more formalized optimization models for this problem and also incorporating survivability capabilities.

ACKNOWLEDGEMENTS

This research has been supported by the US Department of Energy Office of Science and the National Science Foundation. The authors are very grateful to these sponsors for their generous support.

REFERENCES

[1] R. Guerin, A. Orda, "Networks with advance reservations: The routing perspective", *IEEE INFOCOM 2000*, TelAviv, Mar 2000.

[2] N. Rao, *et al.*, "Control plane for advance bandwidth scheduling in ultra high-speed networks," *IEEE INFOCOM 2006 High Speed Networks Workshop*, Barcelona, Spain, April 2006.

[3] Yunyue Lin, *et al.*, "On design of scheduling algorithms for advance bandwidth reservation in dedicated networks," *IEEE INFOCOM 2008 High Speed Networks Workshop*, Phoenix, AZ, April 2008, pp.1-6.

[4] Y. Lin, Q. Wu, "Path computation with variable bandwidth for bulk data transfer in high-performance networks," *IEEE INFOCOM 2009 High Speed Networks Workshop*, Rio De Janeiro, Brasil, April 2009.

[5] J. Zheng, H. Mouftah, "Routing and Wavelength assignment for advance reservation in wavelength-routed WDM optical networks," *IEEE ICC 2002*, New York, NY, May 2002.

[6] J. Kuri, *et al*, "Routing and wavelength assignment of scheduled lightpath demands," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 8, October 2003, pp.1231-1240.

[7] T. Li, *et al.*, "On survivable service provisioning in WDM optical networks under a scheduled traffic model," *IEEE GlOBECOM 2005*, St. Louis, MO, December 2005.

[8] C. Xie, *et al.*, "Load-balancing connection rerouting in advance reservation networks", *IEEE Communications Letters*, vol. 14, no. 6, June 2010, pp. 578-580.

[9] C. Xie, H. Alazemi, N. Ghani, "Routing and Scheduling in distributed advance reservation networks," *IEEE GLOBECOM 2010*, Miami, FL, December 2010.

[10] D. Andersen, *et al.*, "Resilient overlay networks," *ACM SOSP 2001*, October 2001, pp. 131-185.

[11] Z. Duan, *et al*,. "Service overlay networks: SLAs, QoS, and bandwidth provisioning," *IEEE/ACM Transactions on Networking*, vol. 11, no. 6, December 2003, pp. 870-883.

[12] Y. Zhu, M. Ammar, "Algorithms for Assigning substrate network resources to virtual network components," *IEEE INFOCOM 2006*, Barcelona, Spain, April 2006.

[13] C. Xie, *et al.*, "Multi-point Ethernet over next-generation SONET/SDH", *IEEE ICC 2009*, Dresden, June 2009.

[14] C. Xie, *et al.*, "Traffic engineering for ethernet over SONET/SDH (EoS): Advances and frontiers", *IEEE Network*, vol. 23, no. 3. May/June 2009, pp. 18-25.

[15] R. Dutta, G. Rouskas, "A Survey of Virtual topology design algorithms for wavelength routed optical networks," *Optical Networks*, vol. 1, no. 1, January 2000, pp. 73-89.

[16] Inet topology generator, see http://topology.eecs.umich.edu/inet/.