Transactions of the VŠB – Technical University of Ostrava, Mechanical Series
No. 2, 2010, vol. LVI
article No. 1803

Rudolf VOLNER[*]

VIRTUAL AVIATION DATA NETWORKS - DYNAMIC CONSTRUCTION

VIRTUÁLNÍ LETECKÁ DATOVÁ SÍŤ – DYNAMICKÉ ŘEŠENÍ

Abstract

Based virtual aviation data networks (VADNs) are designed to provide services with security and QoS comparable to that of a private network. Traditional traffic modeling of data sources assumed that the inter-arrival times of traffic packets were basically exponential in distribution and independent of one another, which means that the process is memory-less.

Abstrakt

Virtuální datová letecká síť definuje možnosti zvyšování úrovně služeb v závislosti na zachování požadované úrovni bezpečnosti a kvalitě. Článek definuje novou možnost aplikace datové sítě v letectví tzv. virtuální datovou síť (VADN). Jsou popsány základní služby této sítě z ohledem na provoz. V dalších částech jsou pak popsány různé typy virtuální datové sítě s ohledem na definování matematických modelů a s toho plynoucích možností aplikace datové sítě v letectví. Velký důraz je pak kladen na specifikaci dynamické virtuální letecké sítě a její vlastnosti, se snahou vystihnou nejpodstatnější problémy, které jsou spojeny s řešením tohoto typu sítě v provozu, který je aplikován v letecké dopravě.

1 INTRODUCTION

Based virtual aviation data networks (VADNs) are designed to provide services with security and QoS comparable to that of a private network [1, 2]. The QoS guarantee is accomplished through bandwidth specification and reservation in the network. Bandwidth specification of a VADN is provided by the VADN owner and is often done in the form of *service level agreements* (SLAs) that specify the type of services and the amount of bandwidth for each type.

In terms of bandwidth specification and requirement, we can divide VADNs into two types: *pipe-model* VADNs and *hose-model* VADNs. A pipe-model VADN needs to specify the bandwidth requirement between any two endpoints. If the number of endpoints of a VADN is large, a pipe-model VADN is not an efficient solution as it is difficult to precisely predict the bandwidth requirement for each source-destination pair. A hose-model VADN, on the other hand, only needs to specify the amount of ingress and egress traffic (i.e., the amount of traffic that can be sent to and received from the backbone network) at each endpoint. Bandwidth specification is obviously much easier than that of a pipe-model VADN.

Many hose-model VADN provisioning algorithms have been proposed recently. They focus on the bandwidth efficiency in the construction of a single hose-model VADN. If we add the following constraints: single-path, tree-topology, symmetric ingress and egress bandwidth at each endpoint, and infinite link capacity, then a minimum-bandwidth VADN can be constructed in polynomial time.

2 CHARACTERIZATION OF SERVICES, TRAFFIC

Traditional traffic modeling of data sources assumed that the inter-arrival times of traffic packets were basically exponential in distribution and independent of one another, which means that the process is memory-less. However, recent studies of the behaviour of individual multimedia

---

[*] Prof. Ing., Ph.D., VŠB – Technical University of Ostrava, Faculty of Mechanical Engineering, Institute of Transport, Department of Air transport, Dr. Malého 15, 701 00 Ostrava (+420) 59 699 1765, e-mail Rudolf.volner@vsb.cz

sources and system-level activity show that traffic traces are distributed in ways more complex than this.

Our analysis has aimed at improving the best-fitting model for a given traffic scenario when the underlying flow keeps changing over time and space. To be confident that the results are useful a model was sought that:

- was as simple as possible in a computational sense without compromising accuracy,
- had a physical explanation in the network context,
- can be related to real measurements for verification purposes by the operators.

The investigation focused on extensions that could retain tractability, in two steps as described below:

- statistical multiplexing,
- parameterization.

Traffic generation – if the traffic is memory-less, generation of traffic to support the simulations can be achieved simply by a negative exponentially distributed process to specify packet inter-arrival time.

3 SINGLE VADN CONSTRUCTION

Assume that the VADN to be added has endpoints and the ingress and egress bandwidth constraints at the VADN endpoints are given by the following vector

$$H = [(a_1, b_1), \ldots, (a_n, b_n)] \tag{1}$$

Where a_i and b_i represent the VADN's ingress and egress bandwidth requested at node *i*, that is, a_i and b_i are the maximum amount of traffic that end node *i* can send into and receive from the MPLS backbone network.

Compared with a conventional traffic matrix

$$T = \{d_{ij}\} \tag{2}$$

where d_{ij} represents the traffic rate from node *i* to node *j*, a hose-model VADN only provides the row sums

$$\sum_j d_{ij} = a_i \tag{3}$$

a_i is the ingress traffic at node *i* and column sums

$$\sum_i d_{ij} = b_j \tag{4}$$

b_j is the egress traffic at node *j*. Conventional traffic engineering techniques are mostly based on the assumption that

$$T = \{d_{ij}\} \tag{5}$$

is known and cannot be directly applied to a hose-model problem. In the following, we will present several linear programming formulations for constructing a single hose-model VADN.

If we can list every element in *D*, the problem of finding a routing scheme (i.e., x_{ij}^{e}) that minimizes the bandwidth reservation can be formulated as the following linear programming (LP) problem

$$\min \sum_{e \in E} y_e$$

$$\begin{aligned}
&\sum_{e \in G^+(u)} x_{ij}^{e} - \sum_{e \in G^-(u)} x_{ij}^{e} = 0, \quad i, j \in R,\ u \neq i, j \\
&\sum_{e \in G^+(u)} x_{ij}^{e} - \sum_{e \in G^-(u)} x_{ij}^{e} = 1, \quad i, j \in R,\ u = i \\
&\sum_{e \in G^+(u)} x_{ij}^{e} - \sum_{e \in G^-(u)} x_{ij}^{e} = -1, \quad i, j \in R,\ u = j \\
&\sum_{i, j \in R} x_{ij}^{e} d_{ij} \leq y_e, \quad e \in E,\ T \in D \\
&0 \leq y_e \leq c_e, \quad e \in E \\
&0 \leq x_{ij}^{e} \leq 1, \quad i, j \in R,\ e \in E
\end{aligned} \tag{6}$$

The approach outlined in formulation (6) has one problem - elements in D are too numerous to list. The problem is solved by the following property.

This linear programming formulation has a polynomial number of variables and constraints. Once we have x_{ij}^{e}, the set of paths and the load splitting ratios among the paths can be obtained

4 DYNAMIC VADN CONSTRUCTION

VADNs come and go. The dynamics of adding and deleting connections can have a significant impact on the scalability of the network. Current schemes for dynamic VADN construction are based on the constraint-based-routing framework where we first remove the bandwidth reserved for the existing VADNs before creating paths and reserving bandwidth for the new VADN. There are several problems with this approach:

- First, finding the optimal paths for a new VPN is not a trivial task and can be time consuming,
- Second, the number of paths inside the network grows with the number of VADNs,
- Third, each edge router needs to maintain the state information (like splitting ratios) of each individual VADN.

Because the number of VADNs can be very large in a high-speed network, maintaining the state information of each VADN can create a scalability problem. In the following, we solve the problem with a different approach.

4.1 Framework

We propose a new framework for constructing dynamic VADNs. We call a network non-blocking if none of its internal links will ever experience congestion so long as the total ingress and egress traffic does not exceed some specified constraints. Paths inside a non-blocking network are preset up. Whether to accept a new VADN is reduced to checking if the ingress and egress points have enough bandwidth for accepting the VADN. There is no need to check the paths inside the network or to re-compute the internal paths. This avoids the scalability problem in the constraint-based-routing approach.

To design a non-blocking network, we need to determine how much traffic can be admitted by each edge router of the network. Let $\left(q\tilde{a}_i, q\tilde{b}_i\right)$represent the maximum amount of ingress and egress traffic allowed to enter and leave the network at the edge router *i*, where $\tilde{a}_i$ and $\tilde{b}_i$ are constants describing the degree of unevenness of traffic patterns in the network and q is a parameter to be maximized in our design. For example, suppose $\left(\tilde{a}_1 = 5,\ \tilde{b}_1 = 5\right)$ and $\left(\tilde{a}_2 = 15,\ \tilde{b}_2 = 15\right)$, then the traffic allowed at edge router 2 is three times that of router 1.

The preference parameters $\tilde{a}_i$ and $\tilde{b}_i$ are provided by network planners who may obtain these values based on past traffic demands (if there is no prior information about the network, we can simply assume the same value for all $\tilde{a}_i$ and $\tilde{b}_i$). Note that only the relative—not absolute—magnitudes of $\tilde{a}_i$ and $\tilde{b}_i$ have significance as the real amount of admissible traffic is determined by q. As shown below, to determine $\left(q\tilde{a}_i, q\tilde{b}_i\right)$, we also determine the paths between any two edge routers and their load distribution ratios. These paths will be pre-set up. When a new hose-model VADN arrives, we only verify if the ingress and the egress edge routers' utilization is below the constraint specified by $\left(q\tilde{a}_i, q\tilde{b}_i\right)$. If so, accept the VADN; otherwise, reject it.

4.2 Optimal Routing in Non-blocking Networks

We present two algorithms for maximizing q. The *link congestion ratio* is defined as the ratio of the amount of traffic routed through a link over the link's capacity. The *network congestion ratio*, denoted by *r*, is defined as the maximum of all link congestion ratios of the network. If $r\langle 1$, the network has no congestion because traffic routed through any link is below the link capacity.

We use the concept of congestion ratio to maximize. We first assume the ingress and the egress traffic constraints at edge router *i* are $\left(\tilde{a}_i, \tilde{b}_i\right)$ and then compute the congestion ratio *r*. If we change the ingress and egress traffic to $\left(\left(\frac{\tilde{a}_i}{r}\right),\left(\frac{\tilde{b}_i}{r}\right)\right)$, the congestion ratio of the network will be ≤ 1. The maximum admissible amount of traffic at edge router *i* is thus $\left(\left(\frac{\tilde{a}_i}{r}\right),\left(\frac{\tilde{b}_i}{r}\right)\right)$. The problem of maximizing q is now converted to the minimization of *r* as $q = \frac{1}{r}$.

We can use a linear programming formulation to solve the problem of minimizing *r*. Note that in the following formulation, we assume the traffic constraint at edge router *i* is $\left(\tilde{a}_i, \tilde{b}_i\right)$

$$\begin{aligned}
&\min x\\
&\sum_{e\in r^{+}(u)} x_{ij}^{e} - \sum_{e\in r^{-}(u)} x_{ij}^{e} = 0, \quad i, j \in Q,\ u \neq i, j\\
&\sum_{e\in r^{+}(u)} x_{ij}^{e} - \sum_{e\in r^{-}(u)} x_{ij}^{e} = 1, \quad i, j \in Q,\ u = i\\
&\sum_{e\in r^{+}(u)} x_{ij}^{e} - \sum_{e\in r^{-}(u)} x_{ij}^{e} = -1, \quad i, j \in Q,\ u = j\\
&\sum_{i,j\in Q} x_{ij}^{e} d_{ij} \leq c_e.r, \quad e \in E,\ T \in D\\
&0 \leq x_{ij}^{e} \leq 1, \quad e \in E, \quad i, j \in Q
\end{aligned}$$

$$r \geq 0 \tag{7}$$

where (7e) is the bandwidth constraint. Again, constraint (7e) needs to include all valid's. Following a similar argument as in Property 1, formulation (7) can be converted to a polynomial size LP formulation as follows

$$\begin{aligned}
&\min r\\
&\sum_{e \in r^{+}(u)} x_{ij}^{e} - \sum_{e \in r^{-}(u)} x_{ij}^{e} = 0, \quad i, j \in Q,\ u \neq i, j\\
&\sum_{e \in r^{+}(u)} x_{ij}^{e} - \sum_{e \in r^{-}(u)} x_{ij}^{e} = 1, \quad i, j \in Q,\ u = i\\
&\sum_{e \in r^{+}(u)} x_{ij}^{e} - \sum_{e \in r^{-}(u)} x_{ij}^{e} = -1, \quad i, j \in Q,\ u = j\\
&\sum_{i \in Q} \tilde{a} p_{e}(i) + \sum_{i \in Q} \tilde{b}_{i} l_{e}(i) \leq c_{e}.r, \quad e \in E\\
&x_{ij}^{e} \leq p_{e}(i) + l_{e}(j), \quad i, j \in,\ e \in E\\
&0 \leq x_{ij}^{e} \leq 1, \quad i, j \in Q,\ e \in E\\
&p_{e}(i),\ l_{e}(i) \geq 0, \quad i \in Q,\ e \in E\\
&r \geq 0
\end{aligned} \tag{8}$$

The complexity of (8) is much higher than that of (6) due to the hard constraint (8e).

4.3 Decomposition Algorithm

Solving a large-scale LP problem is usually time consuming. However, we can exploit the underlying network flow structure in formulation (8) and convert it into a sub-gradient iterative search problem. In each step, we only need to solve a maximum-flow problem for which very fast algorithms are available. Let $f_{ij}^{e} = x_{ij}^{e} q$. Since $q = \frac{1}{r}$, (8) can be transformed into

$$\begin{aligned}
&\max q\\
&\sum_{e \in r^{+}(u)} f_{ij}^{e} - \sum_{e \in r^{-}(u)} f_{ij}^{e} = 0, \quad i, j \in Q,\ u \neq i, j\\
&\sum_{e \in r^{+}(u)} f_{ij}^{e} - \sum_{e \in r^{-}(u)} f_{ij}^{e} = q, \quad i, j \in Q,\ u = i\\
&\sum_{e \in r^{+}(u)} f_{ij}^{e} - \sum_{e \in r^{-}(u)} f_{ij}^{e} = -q, \quad i, j \in Q,\ u = j\\
&\sum_{i \in Q} \tilde{a} p_{e}(i) + \sum_{i \in Q} \tilde{b}_{i} l_{e}(i) \leq c_{e}, \quad e \in E\\
&f_{ij}^{e} \leq p_{e}(i) + l_{e}(j), \quad i, j \in Q,\ e \in E\\
&f, q, p, l \geq 0,
\end{aligned} \tag{9}$$

4.4 Setting a Limit on the Number of Paths

In the previous discussion, there is no restriction on the number of paths between each source–destination pair. Our results indicate that most traffic loads are distributed among a small number of paths. It is desirable to set a path limit to reduce the external label table size and to simplify load distribution. In the following, we show how to impose a path-number limit.

Let *L* be the number of paths allowed for each source–destination pair. We then select the most loaded *L* paths among those generated by the decomposition algorithm. Given the set of paths, we re-compute q by solving a linear programming problem similar to that given in formulation (9). The difference is that path-flow representation, instead of link-flow representation, must be used in the formulation. Let P_{ij} be the set of paths between the node pair (i, j), $x(p)$ the amount of flow sent on path *p*, and $d_e(p)$ a link-path indicator variable, that is, $d_e(p)$ equals 1 if link *e* is contained in the path *p*, and 0 otherwise. Following a similar formulation of (8), we can use the following linear programming formulation to find the maximum q

$$\begin{aligned} &\max q \\ &\sum_{p\in P_{ij}} x(p) = q, \quad i, j \in Q \\ &\sum_{i\in Q} \tilde{a} p_e(i) + \sum_{i\in Q} \tilde{b}_i l_e(i) \le c_e, \quad e \in E \\ &\sum_{p\in P_{ij}} d_e(p) x(p) \le p_e(i) + l_e(j), \quad i, j \in Q, e \in E \\ &x, p, l, q \ge 0 \end{aligned} \tag{10}$$

5 CONCLUSION

Provisioning VADNs in the hose model differs from that in the traditional pipe model in that the traffic demand matrix is unknown and only the maximum bandwidths of the traffic each VADN endpoint can send into and receive from the network are given. Determining optimal routing and bandwidth reservation for VADNs in the hose model is a challenging problem because of the uncertainty of the point-to-point load distribution. Previous research work focuses on routing and bandwidth provisioning for single VADN.

The analytical techniques developed in the paper are general and can be used to tackle other network problems when traffic uncertainty is inherent. We will explore this issue in our future research.

REFERENCES

[1] ROSEN, E., et al.. Multiprotocol label switching architecture, *RFC 3031*, January 2001.

[2] ROSEN, E., REKHTER, Y. BGP/MPLS VPNs, *RFC 2547*, Mar. 1999.

[3] JÜTTNER, A., SZABO, I., SZENTESI, A. On bandwidth efficiency of the hose resource management model in virtual private networks, *Proceedings of. IEEE INFOCOM 2003*, San Francisco, April 2003, pp. 386–395.

[4] DUFFIELD, N., G., GOYAL, P., GREENBERG, A., MISHRA, P., RAMAKRISHNAN, K., K., DER MERWE, J., E., V. A flexible model for resource management in virtual private networks, *Proceedings of ACM SIGCOMM*, San Diego, CA, August 1999, pp. 95–108.

[5] KUMAR, A., RASTOGI, R., SILBERSCHATZ, A.,YENER, B. Algorithms for provisioning virtual private networks in the hose model, *Proceedings of ACM SIGCOMM*, Cambridge, MA, August 2001, pp. 135–146.

[6] GUPTA, A., KLEINBERG, J., KUMAR, A., RASTOGI, R., YENER, B. Provisioning a virtual private network: A network design problem for multicommodity flow, *Proceedings of ACM STOC*, 2001, pp. 389–398.

[7] VOLNER, R., POUŠEK, L. Wireless Biomedical Home Security Network – architecture and modelling", *38th Annual 2004 International Carnahan Conference on Security Technology*, October 2004 Albuquerque, New Mexico, USA, pp. 69 – 76, IEEE Catalog Number

04CH37572, ISBN 0-7803-8506 - 3

[8] VOLNER, R., POUŠEK, L. Inteligence Security Home Network, *37th Annual 2003 International Carnahan Conference on Security Technology*, October 2003 Taipei, Taiwan, pp. 30 – 37, IEEE Catalog Number 03CH37458 , ISBN 0-7803-7882-2,

[9] VOLNER, R. Inteligence CATV – Traffic models, Design and Analysis, *International Conference on Computer, Communication and Control Technologies CCCT'03 and The 9th International Conference on Information Systems Analysis and Synthesis ISAS 03*, Proceeding volume IV, July 2003, Orlando, Florida, USA, pp. 340 – 345, ISBN980-6560-05-1, CD - ISBN 980-6560-10-8,

[10] TICHÁ, D. A Sensitivity Approach in Digital Filter Design, *Proceedings of 3rd International Workshop Digital Technologies 2006*, Žilina, 2006, ISBN 80-8070-637-9

[11] VOLNER, R., BOREŠ, P. Aviation Data Networks, *Electronics and Electrical Engineering* N^{o} 7 (63), Kaunas University of Technology, Academy of Sciences of Lithuana, Vilnius Gedimas Technical University, Riga Technical University, Tallinn Technical University, Kaunas 2005, pp. 22-26, ISSN 1392-1215