A Practical Approach to VPN Resource Management using a Dynamic Hose Model

Christian Müller
Institute of Communication Networks
and Computer Engineering
University of Stuttgart, Germany
Email: mueller@ikr.uni-stuttgart.de

Emmanuel Dotaro and Dimitri Papadimitriou
Alcatel Research & Innovation
Marcoussis, France
Email: emmanuel.dotaro@alcatel.fr,
dimitri.papadimitriou@alcatel.be

***Abstract*— Multipoint Ethernet Services and Virtual Private LAN Services are an active field of both research and standardization today. A high degree of automation and reduced customer-provider interaction, together with efficient resource management schemes and a guaranteed QoS level are key requirements for these new services, which constitute challenging problems to service providers. Especially the integration into future (G)MPLS networks is an urging need to many operators. In this article, we evaluate VPN resource management schemes under realistic constraints with respect to their applicability to present service developments for (G)MPLS networks. Based on this evaluation, we propose and discuss a new dynamic hose model-based capacity management scheme, which explicitly targets at keeping signaling load and measurement complexity low. Finally, we prove the applicability of the novel approach by means of simulation in a representative case study.**

I. Introduction

Virtual Private Networks, in particular Transparent LAN Services and Multipoint Ethernet Services, are presently being developed with enormous effort for a number of reasons. On the one hand, service providers are looking for new revenue-generating services to better exploit resources in their networks. On the other hand, customers request more and more flexible services and short provisioning times to meet rapidly changing business relations, which in-turn lead to changes in a company's communication patterns. It is expected that sites will join and leave VPNs at relatively short timescales, compared to today's provisioning times in the order of several days or weeks. To keep operation and management cost low, operators would like such new services to support a high degree of automation and reduced customer-provider interaction. For example, joining or leaving VPNs should be a largely automated process, possibly using policy-based configuration mechanisms on the provider side.

Given that technologies like Multi-Protocol Label Switching (MPLS) and Generalized MPLS (GMPLS) are gaining ground [1] and Ethernet is moving towards the MAN and WAN domain [2], we can observe a number of ongoing activities in standardization bodies, which try to stay abreast of changes. Ethernet label switching or Virtual Private LAN Service (VPLS) development within the IETF [3], Provider Backbone Bridges in the IEEE [4] and the Metro Ethernet Services developed by the MEF [5] are just a few examples which head in this direction. Although new requirements have to be derived from such an environment, which restrict the applicability of known VPN resource management schemes, this has not been addressed in these standards so far.

The remainder of this paper is organized as follows: Section II starts with the description of our VPN reference architecture in (G)MPLS networks. In section III, we discuss the applicability of known resource management schemes with respect to this architecture. Based on this, we present and evaluate a new flexible resource management scheme in sections IV and V, respectively.

II. VPN Reference architecture

We consider a reference model of a provider-provisioned VPN as it is depicted in Fig. 1. This model is a generalized view of the Layer 2 VPNs currently being developed within the IETF [3] and also applies to the service architecture specified by the MEF [5]. Multiple customer networks are connected to the provider network over an Attachment Circuit (AC) to so-called Provider Edge devices (PE). From a provider's point of view, the first device under customer premises is denoted as Customer Edge device (CE), which usually is nothing but a standard layer 2 switching or terminating device. All PEs belonging to a customer VPN are fully meshed using VPN tunnels. A *VPN tunnel* is independent of whether the VPN is realized using Pseudo-Wires [6] or by means of Ethernet label switching techniques [7]. It is also independent of the transport network technology being either label switching or connectionless packet forwarding. Within the network, we

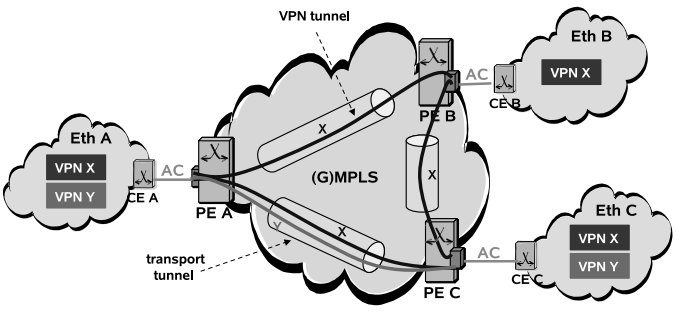

Fig. 1. (G)MPLS-based VPN scenario

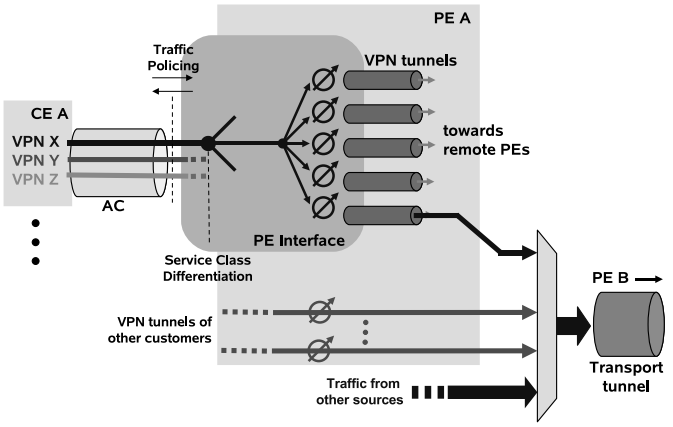

Fig. 2. PE node reference model

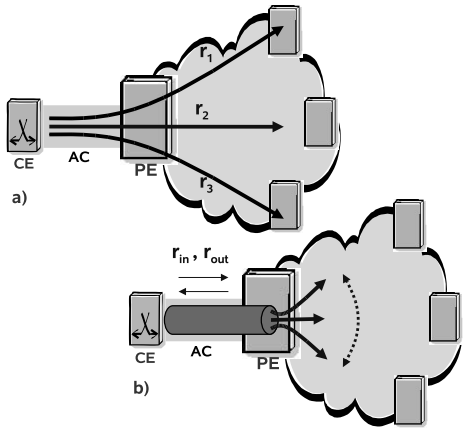

Fig. 3. Service interfaces: a) customer-pipes and b) hose model

assume that VPN tunnels are further aggregated into *transport tunnels*, which are more coarse-grained tunnels used to save resources and to hide the complexity of a VPN service from core devices. For example, in an MPLS network, these coarse-grained tunnels are MPLS LSPs. In case of Pseudo-Wires, they are required to begin and end on the respective PEs of a pair of VPN service endpoints, whereas this requirement is less stringent in a layer 2-switched GMPLS region.

A customer network can be part of multiple VPNs, which share a common AC but are transported over different VPN tunnels inside the provider network. This is visualized in the detail view of a CE-PE interface in Fig. 2. We suppose traffic policing functions to be performed on a per VPN level and possibly also on a per interface level, in which case an interface might be the terminating point of multiple services instances. This means that, for reasons of higher flexibility of the service interface, hose parameters (see section III) might be provided for a set of VPNs only.

Besides the VPN architecture itself, several more characteristics of this reference model have an impact on service provisioning and operation. (G)MPLS networks rely on a distributed control plane, in which reachability, topology and traffic engineering information are distributed by link-state routing protocols to all routers in the network [8]. Service-specific parameters or interface configuration details however are not part of this information, given that routing protocols should not be overloaded with non-routing information. Furthermore, (G)MPLS does not require a network management plane for its operation. In our reference architecture, we assume the operator to employ some console-based or SNMP-assisted network management, as it has been outlined in [1].

Depending on the used quality of service framework, different parameters are employed for resource reservation requests, as for example an extended token bucket specification in the controlled load service [9]. These parameters are used by local resource management to exploit statistical multiplexing gains among reservations on the same link. However, mapping rules of resource reservation values to these signaling parameters are out of scope of this article.

III. VPN RESOURCE MANAGEMENT CONCEPTS

A straightforward VPN resource management concept is the so-called *customer-pipes model*, in which operators reserve network resources exactly as specified by a customer-provided traffic matrix (Fig. 3a). Other approaches try to cope with dynamics in traffic patterns by allowing customers to continuously adjust the bandwidth allocated to their VPN [10].

It has already been mentioned that novel VPN services will have to face dynamics in customer behavior at an increasing rate. In such a rapidly changing environment, it is difficult to accurately describe and predict communication patterns for a given VPN. We believe that customers cannot provide, and most likely also do not want to provide a detailed traffic matrix specifying the traffic volume being exchanged between any two sites of their VPN. Furthermore, in order to keep customer-provider interaction low, operators do not want customers to interfere with their service provisioning too frequently. Consequently, the resource management models described above are not applicable and we focus on resource management approaches based on the so-called *hose model*.

The hose model was first presented in [11], its principle is depicted in Fig. 3b. Customers only provide bounds for the maximum amount of traffic which is injected into or received from the provider network ($r_{\mathrm{in}}, r_{\mathrm{out}}$), but they do not provide any information about its spatial distribution over remote service endpoints. Hence, in addition to temporal load variations, providers have to deal with uncertainty about the traffic's spatial distribution and provide enough resources to accommodate for the worst case traffic split. The authors in [11]–[15] investigated various ways to face this challenge, which can be classified in static and dynamic approaches.

The static provider-pipes approach is the most simple of the static schemes, consisting of reservations at the hose's peak rate between any two service endpoints. Obviously, efficiency is very low and the over-provisioning factor, i.e. the bandwidth requirement compared to the corresponding customer-pipes dimensioning, has been found to increase linearly with the number of nodes in the network, which makes this approach inappropriate for practical application [12].

More efficient static schemes make use of shared reservations for tunnels with a common service endpoint or, more generally, resources are shared among tunnels of the same VPN on common links anywhere in the network. These resource-sharing approaches differ with respect to a number of parameters, like for example support of symmetric or asymmetric interface parameterization, shortest-path, explicit or multi-path routing patterns and the amount of information needed for service provisioning. Some approaches rely on sink-rooted or source-rooted trees on the respective service endpoints, others try to compute a near-optimum topology for the entire VPN. In [13], [14], the authors showed that the computation of such resource sharing topologies can be computationally hard, depending on the type of routing and whether hose parameters are symmetric or not. The authors in [15] proposed an algorithm to calculate multi-path topologies and compared the performance of several approximation algorithms. They found that running times of these topology computation algorithms increase very quickly with the number of nodes and can be in the order of minutes for large networks. In addition, computational complexity is likely to be further increased by the flexible service interface concept described in section II.

In [12], it has been shown that in order to achieve reasonably low over-provisioning factors, the computation of a tree-structured resource-sharing topology for the whole VPN using explicit routing is the only viable candidate among the statically provisioned models without multi-path routing.

In general, these computations require a global view of the VPN and the parameters on the respective service endpoints. Given that the distribution of VPN-related information in a (G)MPLS network is limited, computations of VPN-wide resource-sharing topologies are infeasible as there is no such device which would be able to perform these calculations and to accordingly set up LSPs. It might be argued that network management could perform this task. However, we believe present centralized network management mechanisms would not scale well when it comes to large-scale deployments of VPN services where customers modify, join or leave VPNs at a much shorter timescale than it is the case today.

For some particular static resource-sharing approaches, several other constraints have to be noted. For example, routing patterns like multi-path routing can decrease computational complexity, but they are much more demanding in terms of label resources which might also be a non-negligible factor. As for the resource-sharing approaches based on source-rooted or sink-rooted trees, despite their reportedly poor performance, source-rooted trees are difficult to realize. Point-to-multipoint LSPs set up by RSVP-TE are inappropriate here, as this would imply traffic replication at branching nodes. Sink-rooted trees effectively correspond to multipoint-to-point LSPs, which might conflict with MAC address learning mechanisms on edge devices required by many L2VPN realizations. If devices are configured to assign separate labels per sender, this problem might be circumvented, but then again label resources are likely to be the limiting factor. For tree-structured and multi-path routing schemes, frame forwarding decisions would have to be taken not just on edge devices but also at branching points inside the core network, which is contradictory to the VPN development in [3].

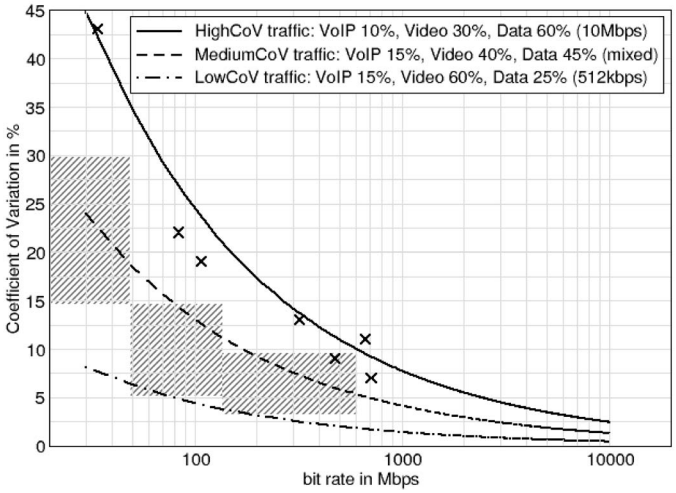

Fig. 4. Parameter range for CoV estimation of the aggregated traffic

As for the dynamic approaches, measurement-based concepts have already been proposed in [11], which require measurements of the mean bit rate and the variance on access links, respectively on every link in the network. These values have been used to compute the actual bandwidth requirement according to a so-called *Local Gaussian Predictor*, which is an algorithm similar to the equivalent capacity formula presented in [16]. Both formulas constitute open-loop zero loss adaptive bandwidth control algorithms and are based on the assumption that the rate distribution is approximately Gaussian [17]. The dynamic resource management schemes we have found in previous work usually require rather complex measurements, either with respect to the number of measurement points in the network or concerning the parameters which have to be measured, or both. Furthermore, the impact of reservation update signaling of these approaches on the control plane and the amount of additional signaling traffic are widely ignored.

Combinations of static resource-sharing and dynamic measurement-based resizing approaches have been investigated as well. However, as long as we do not make use of a VPN-wide optimized resource-sharing topology, a combination of the two approaches does hardly provide any benefit over a dynamic provider-pipes concept [11].

Recently, a new resource management concept has been proposed which is called the *point-to-set model* [18]. The major drawback of the traditional *customer-pipes model* to require detailed information on traffic distribution is relaxed to a specification of a set of destinations and the mean and variance of the traffic fraction to each of these service endpoints. However, this specification still trades off flexibility of the customer's traffic patterns against resource efficiency of the VPN realization in the provider network.

IV. Dynamic resource management for VPNs in (G)MPLS networks

In order to compensate for the shortcomings mentioned in the previous section, we present a resource management scheme which is derived from the provider-pipes approach with dynamic resizing, as it has been introduced in [11]. Our modifications to the original approach head for two main objectives in order to improve applicability in real networks: to keep measurement complexity low and to limit control plane impact due to reservation update signaling.

A. Measurement Complexity

Measurements in our proposal are restricted to mean bit rate measurements on edge nodes of a VPN, averaged over intervals of several minutes length. The respective measurement points are indicated by measurement symbols in Fig. 2. Obviously, the mean bit rate is not sufficient to derive resource reservation settings for VPN tunnels under bursty data traffic. Instead of measuring the coefficient of variation (CoV) of the aggregated traffic in realtime, we make use of an estimation. While it is difficult to accurately estimate the CoV, we state that it is possible to define a range of values in which the CoV is to be found.

For our study, we based our estimates on measurements in real networks. The data values (X) in Fig. 4 show measurements of the CoV reported by [19], while the hatched areas are extracted from [20]. We combined these measurements with curves of representative traffic composites consisting of varying proportions of the traffic types Voice-over-IP, Video and Data traffic, as it can also be seen from Fig. 4. With the intention of being conservative in our assumptions about the characteristics of a VPN flow, we assumed the aggregated traffic to exhibit very bursty traffic properties. Our CoV estimation thus follows the top-most curve of Fig. 4, denoted as *HighCoV* traffic. In a practical application, an operator might conduct test measurements in his own network to determine the corresponding range of values with higher accuracy.

In order to obtain a meaningful bandwidth reservation value, the estimated CoV together with the measured mean bit rate are fed into an equivalent capacity formula. We apply the formula denoted as stationary approximation in [16], given that it has been found to provide conservative, but satisfying results over a large parameter range [21].

B. Reservation Update Signaling

This procedure does not yet reduce the rate of resource reservation updates. We thus propose a combination with a filter mechanism, as it is depicted in Fig. 5. The mode of operation is as follows: bit rate measurements are passed on to a *Filter* unit, where they are processed by a filter algorithm and discretized with respect to a given bandwidth reservation granularity. If the filtered mean bit rate value is sufficiently different from the mean rate of the current reservation, the calculation of a new bandwidth reservation is triggered. The *Resource Reservation Calculation* unit determines the VPN flow's current equivalent capacity as it has been outlined above. The new reservation value is passed on to the *Signaling Controller* unit which, in a GMPLS node, maintains RSVP-TE signaling sessions with remote PEs. The following filters have been studied in our work:

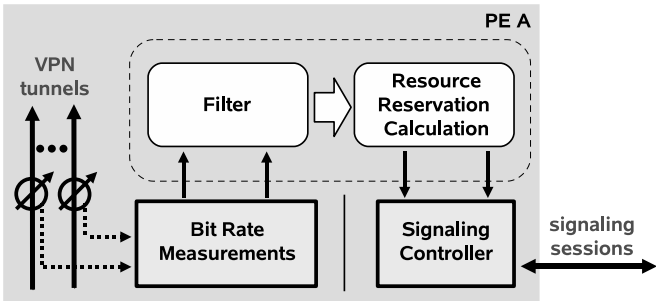

Fig. 5. Functional Architecture

- A *ThresholdHysteresis* filter, represented by a step function with the reservation granularity as increments and combined with a hysteresis of one half of the reservation granularity. For our simulation experiments presented in section V, the reservation granularity was either set to 5 or 10 Mbps.
- An extension of the *ThresholdHysteresis* filter is denoted as *DelayedThreshXmin*, where a holding time of *X* minutes is introduced which maintains a given reservation level even if a decreasing bandwidth demand is detected. The number of reservation updates is reduced at the expense of bandwidth efficiency, but it remains equally responsive to increasing demand as the first filter. We considered holding times from 10 to 60 minutes in our studies so far.
- An Exponential Moving Average algorithm which calculates the mean bit rate to $e_t = (1-\alpha)e_{t-1} + \alpha\hat{s}_t$, where e_t is an interpolation between the previous filter outcome e_{t-1} and the current observation $\hat{s}_t$. In combination with a hysteresis, this filter is denoted as *EMAHysteresis*$[\alpha]$. The factor α has been chosen to 0.3 respectively 0.7, to put more or less weight to the measurement history.

V. Case study

We quantified the achievable bandwidth savings and compared our dynamic scheme to a static provider-pipes approach. The effect on the control plane has been investigated by the mean inter-update time of a given VPN tunnel over 24h.

A. Simulation Setup

Simulation setup consists of 10 VPN tunnels sharing the same destination PE. We assume these 10 end-to-end tunnels to be nested into a more coarse-grained transport tunnel as it is shown in Fig. 2. The simulation topology thus is represented by two PE nodes with 10 VPN tunnels in between, aggregated into a transport tunnel. We accounted for statistical multiplexing gains from the aggregation of the multiple VPN tunnels in the network by application of the capacity allocation scheme proposed in [16], where the equivalent capacity of

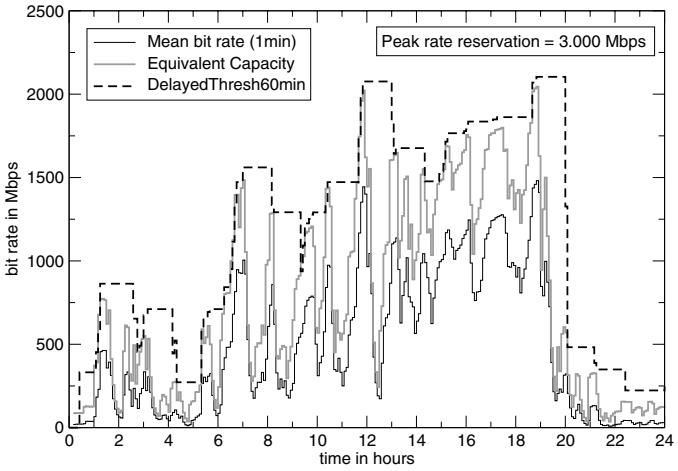

Fig. 6. Resource reservation with matching assumptions on traffic characteristics (*HighCoV* traffic)

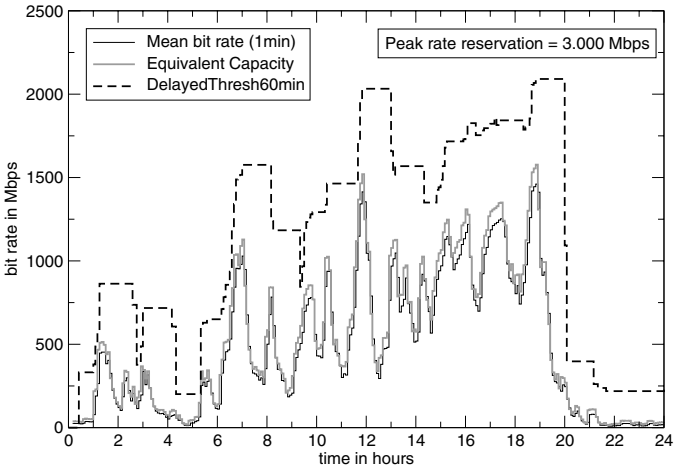

Fig. 7. Over-provisioning for largest deviation of traffic characteristics from CoV estimate (*LowCoV* traffic)

the whole transport tunnel is computed using the parameters mean bit rate and CoV of all currently active VPN tunnel reservations.

Simulations in our case study are conducted on a flow-level, where flows characterize user sessions of either Voice-over-IP (VoIP), Video or Data traffic. There is thus always a random number of active flows in the system, each of them having specific characteristics as for example a certain mean and peak bit rate. The VoIP traffic follows a model of a G.771 coded stream with voice activity detection. To account for video conferencing traffic, the Video class parameters are derived from measurements of a H.263 encoded video stream. The Data traffic class is based on a web traffic model with ON/OFF sources. While the peak bit rate of the previous two classes is usually limited by the application itself, this is different for web traffic, where the peak rate is limited by the access rate currently available to the user. To account for the varying burstiness, web traffic is modeled with peak data rates from 512 kbps to 10 Mbps.

The arrival process of new flows into the system is modeled as a finite source system with negative-exponential interarrival times, in which the number of sources is varied according to a diurnal traffic profile. The profiles have been extracted from measurements on a Fast Ethernet link connecting two large campus local area networks. The average traffic over a 24h cycle for each of the 10 VPN tunnels has been set to 60 Mbps. The equivalent capacity formula is applied such that a target overflow probability of 10^{-8} is met. At peak, this translates to a bandwidth requirement of roughly 300 Mbps for each of the VPN tunnels if *HighCoV* traffic properties are assumed (Fig. 4). We consider this value to be the upper bound for the bandwidth which can be assigned to a single VPN, provided by traffic policing specifications. The bandwidth requirement calculated for a single VPN tunnel is thus not allowed to exceed this value.

B. Principal Behavior

Fig. 6 depicts a simulation run for a reservation granularity of 10 Mbps and a measurement interval length of 5 min. The thin black line shows the sum of the mean bit rates of all user sessions, whereas the thicker grey line represents the corresponding equivalent capacity of the sum of all VPN traffic on this transport tunnel. The resulting reservation envelope is shown as the dashed black line in the figure. This reservation state is computed as the equivalent capacity of all individual VPN tunnels using the mean bit rate and the estimated CoV of the respective reservations. It is thus not the arithmetic sum of all individual equivalent capacities. The applied filter is the *DelayedThreshXmin* algorithm with an X=60 min. holding time, which can nicely be seen from the curve's step function character.

In a static peak rate provisioning scheme, the amount of reserved bandwidth would be a constant value of 10 times the 300 Mbps of a single VPN tunnel. Although this value is solely theoretical, it serves as a reference value independent of the traffic profile and has also been used in [11], [18]. For the traffic profile in Fig. 6, the lower bound for the bandwidth requirement can be given to 25.4% of this reference value. This value would be reached if the reservation followed the actually required equivalent capacity at every instant.

The average bandwidth requirement in our dynamic reservation scheme accounts for only 37% of the reference value, although the filter is very conservative in nature. This resource efficiency is due to two effects: On the one hand, there are statistical multiplexing gains from the aggregation of the VPN tunnels, which are not exploited by the static peak rate provisioning scheme. On the other hand, repeated updates of the resource reservation allow us to keep track of diurnal load variations. If we imagine the same assumptions on traffic characteristics to be applied also for static provisioning (i.e., a mean bit rate and a certain CoV), we could benefit from statistical multiplexing effects also in the static scheme. Considering the offered traffic in the busy hour, the resulting

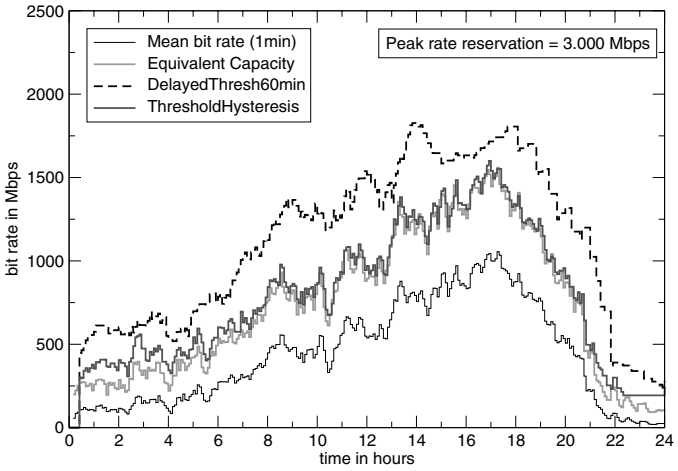

Fig. 8. Over-provisioning due to defensive filter algorithms

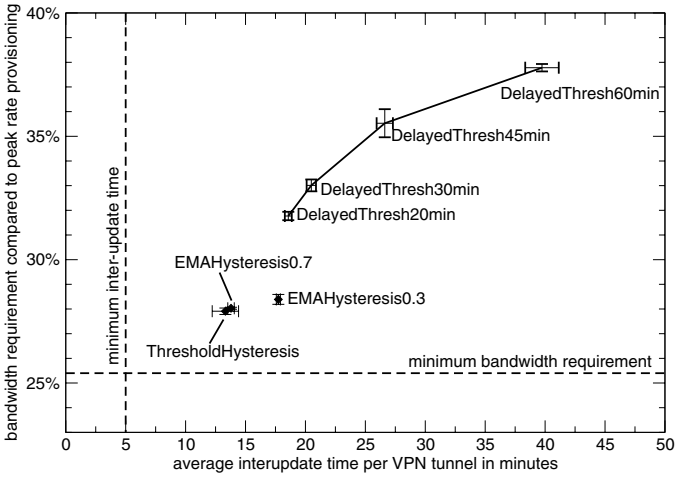

Fig. 9. Trade-off of resource efficiency versus signaling load

reservation would be at 2.154 Gbps for the sample scenario in Fig. 6, while the dynamic reservation scheme still only consumes around 52% of these resources.

Given that real traffic in most cases is much smoother than what has been assumed in the calculation of the bandwidth reservations, exactly the same simulation setup has been examined, now assuming *LowCoV* properties. In consequence, in Fig. 7, we can observe substantial over-provisioning with respect to the actually required capacity for the aggregated traffic. It can thus be argued that if the operator had better information about the kind of traffic being transmitted, this could be used to improve traffic characteristics estimation and in turn lead to increased resource efficiency.

C. Effect of Traffic Profiles on Resource Efficiency

Simulation runs in Fig. 6 and 7 have been conducted with identical diurnal profiles for all 10 VPNs. In order to evaluate a more realistic scenario, the same diurnal profile has been randomly shifted in time by a value in the range of +/-2.5h. Then, phases of high respectively low load do not exactly match anymore among the VPN tunnels, which can be observed from the much smoother profile in Fig. 8. The evaluation has been conducted again by application of the same filter and assuming *HighCoV* traffic properties. Now, over-provisioning in Fig. 8 is mostly caused by effects of the filter algorithm itself, more precisely by the holding time introduced by the *DelayedThreshXmin* algorithm. As the algorithm is individually applied to each VPN tunnel and because the traffic profiles are shifted in time, the dashed line in Fig. 8 does not reflect the step function character of the algorithm anymore. With an average bandwidth requirement of 36% of static peak rate provisioning, resource efficiency is comparable to what has been observed in the previous sample scenarios. Higher gains are achievable if other filter algorithms are chosen which follow the measurements more closely. As an example, the resulting reservation state by application of a simple *ThresholdHysteresis* filter allows to reduce the average bandwidth requirement to 28%. However, this is at the expense of a significantly increased signaling load.

D. Effect of Filters on Signaling Load

We quantified the resulting signaling load by calculation of the mean inter-update time per VPN tunnel. Fig. 9 shows the inter-update times for a range of different filters for the traffic profile used in previous simulation runs. The dashed black lines represent the theoretical minimum of inter-update time and bandwidth requirement, respectively.

While an inter-update time of around 10 to 15 minutes does not constitute an important charge to the control plane for a single signaling session, an operator has to expect hundreds or even thousands of such connections being set up over the same transport tunnel [22]. In this case, load caused by reservation update signaling becomes significant and we have to trade off resource efficiency against signaling load. As it can be seen from Fig. 9, the application of even basic filters can already reduce the inter-update times by a considerable factor, with only around 10% decrease in resource efficiency compared to the static peak rate provisioning. By application of more sophisticated filters, e.g. a threshold-based filter which reacts proactively to an increasing bandwidth demand, a further reduction of inter-update times while maintaining a comparable level of resource efficiency might be achieved.

VI. Conclusion and Final Remarks

In this article, we presented an evaluation of VPN resource management schemes with respect to their applicability to present L2VPN developments in (G)MPLS networks. Important practical constraints such as the minimization of measurement efforts or the impact of resource reservation signaling on the control plane have not been properly accounted for in most previous work. Taking this into account, we proposed a practical dynamic resource management scheme under more realistic constraints, whose new key characteristics are the estimation of the second moment of the aggregated traffic and the application of filters to keep control plane impact low.

The proposed model is highly flexible, allowing customer networks to join or leave a VPN at any time. A first case study showed that even with such a basic and rather conservative approach, considerable bandwidth savings can be achieved. At the same time, signaling load caused by reservation updates can be limited to to what we believe to be a tolerable level.

However, several issues remain which require further investigation. In order to quantify bandwidth savings and related signaling load more thoroughly, a mix of traffic profiles from different scenarios has to be investigated. A direct comparison to the static tree-structured resource-sharing approach presented in [12], [13] in terms of overprovisioning factors would be very interesting, but demands for concrete network topologies and VPN traffic matrices and is thus left for further study.

Further empirical studies are required regarding the estimation of the coefficient of variation, especially in case the VPN traffic is generated by a relatively small number of high capacity sources instead of a local area network with a large number of users. While Guérin's equivalent capacity formula has been found to perform well even in case of self-similar traffic [21], it has been reported that some parameter tuning is required to actually meet the desired performance targets. The fact that in our case input parameters are derived from measurements further motivates studies on the accuracy which can be achieved here, possibly also by application of other equivalent capacity formulas.

With a minimum inter-update time of 5 min., reactivity to load changes in a customer network is limited. Nevertheless, over-reservation on neighbor VPN tunnels due to conservative estimation of traffic properties and defensive filter algorithms can level out the impact of rather long response times. A quantification of this effect also remains for further work.

ACKNOWLEDGMENT

The authors would like to thank Christoph M. Gauger and Sebastian Gunreben for many valuable discussions and suggestions which helped to improve the quality of this contribution.

REFERENCES

[1] M. J. Morrow, M. Tatipamula, and A. Farrel, "GMPLS: The promise of the next-generation optical control plane," IEEE Communications Magazine, Special Issue, July 2005.

[2] G. Chiruvolu, A. Ge, D. Elie-Dit-Cosaque, M. Ali, and J. Rouyer, "Issues and approaches on extending ethernet beyond lans," *IEEE Communications Magazine*, vol. 42, p. 2, 2004.

[3] IETF, "IETF working group: Layer 2 virtual private networks," available at: http://www.ietf.org/html.charters/l2vpn-charter.html.

[4] IEEE, "IEEE 802.1ah - provider backbone bridges," available at: http://www.ieee802.org/1/pages/802.1ah.html.

[5] MEF, "Technical specification MEF 10: Ethernet services attributes phase 1," available at http://www.metroethernetforum.org, 2004.

[6] IETF, "IETF working group: Pseudo-wire emulation edge-to-edge," available at http://www.ietf.org/html.charters/pwe3-charter.html.

[7] D. Papadimitriou and J. Choi, "A framework for Generalized MPLS (GMPLS) Ethernet," IETF draft-papadimitriou-ccamp-gmpls-ethernet-framework-00.txt, work in progress, June 2005.

[8] IETF, "Generalized Multi-Protocol Label Switching (GMPLS) Architecture," RFC 3945 (Proposed Standard), Oct. 2004. [Online]. Available: http://www.ietf.org/rfc/rfc3945.txt

[9] IETF, "Specification of the Controlled-Load Network Element Service," IETF RFC 2211 (Proposed Standard), September 1997.

[10] C. Hota, S. K. Jha, and G. Raghurama, "Distributed dynamic resource management in ipvpns to guarantee quality of service," in *Networks, 2004. (ICON 2004). Proceedings. 12th IEEE International Conference on*, vol. 1, 2004, pp. 414–419 vol.1.

[11] N. G. Duffield, P. Goyal, A. G. Greenberg, P. P. Mishra, K. K. Ramakrishnan, and J. E. van der Merive, "A flexible model for resource management in virtual private networks," in *SIGCOMM*, 1999.

[12] A. Juttner, I. Szabo, and A. Szentesi, "On bandwidth efficiency of the hose resource management model in virtual private networks," in *IEEE INFOCOM*, vol. 1, March 2003.

[13] A. Kumar, R. Rastogi, A. Silberschatz, and B. Yener, "Algorithms for provisioning virtual private networks in the hose model," *Networking, IEEE/ACM Transactions on*, vol. 10, no. 4, 2002.

[14] C. Swamy and A. Kumar, "Primal-dual algorithms for connected facility location problems," in *5th Intern. Workshop on Approximation Algorithms for Combinatorial Approximation (APPROX'02)*, 2002.

[15] T. Erlebach and M. Ruegg, "Optimal bandwidth reservation in hose-model VPNs with multipath routing," in *IEEE INFOCOM*, 2004.

[16] R. Guerin, H. Ahmadi, and M. Naghshineh, "Equivalent capacity and its application to bandwidth allocation in high-speed networks," *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 7, 1991.

[17] P. Siripongwutikorn, S. Banerjee, and D. Tipper, "A survey of adaptive bandwidth control algorithms." *IEEE Communications Surveys and Tutorials*, vol. 5, no. 1, 2003.

[18] S. Raghunath and S. Kalyanaraman, "Statistical point-to-set edge-based quality of service provisioning," in *International Workshop on Quality of Future Internet Services, QoFIS 2003*, vol. 2811, 2003.

[19] L. Yao, M. Agapie, J. Ganbar, and M. Doroslovacki, "Long range dependence in internet backbone traffic," in *IEEE International Conference on Communications*, vol. 3, May 2003.

[20] C. Barakat, P. Thiran, G. Iannaccone, C. Diot, and P. Owezarski, "Modeling internet backbone traffic at the flow level," *IEEE Transactions on Signal Processing*, vol. 51, no. 8, August 2003.

[21] S. Bodamer and J. Charzinski, "Evaluation of effective bandwidth schemes for self-similar traffic," in *13th ITC Specialist Seminar on IP Traffic Measurement, Modeling and Management*, 2000.

[22] A. Nagarajan, "Generic Requirements for Provider Provisioned Virtual Private Networks (PPVPN)," RFC 3809 (Informational), June 2004.