An Efficient Process for Estimation of Network Demand for QoS-aware IP Network Planning

Alan Davy, Dmitri Botvich, and Brendan Jennings

Telecommunications Software & Systems Group,
Waterford Institute of Technology,
Waterford, Ireland
{adavy, dbotvich, bjennings}@tssg.org

Abstract. Estimations of network demand are an essential input to the IP network planning process. We present a technique for per traffic class IP network demand estimation based on harnessing information gathered for accounting and charging purposes. This technique represents an efficient use of pre-existing information, is easy to deploy, and, crucially, is highly cost-effective in comparison to traditional direct measurement systems employing dedicated traffic metering hardware. In order to facilitate QoS-aware network planning we also introduce a technique for estimation of QoS related effective bandwidth coefficients via analysis of a relatively small number of packet traces. The combination of the demand and effective bandwidth coefficient estimation techniques provide the basis for an effective, low-cost network planning solution. In this paper we present initial results that validate our contention that network accounting records can be reused to create a QoS aware demand matrix for IP networks.

1. Introduction

An ISP must ensure that network resources are being utilised optimally in order to avoid unnecessary expenditure. Of course, it is also important that the process of planning, deploying and managing network resources is itself cost-effective and does not significantly degrade network performance. Network planning, in particular, is a vital part of any ISP's business. Currently, network planning typically involves the use of dedicated metering hardware to gather and collate large amounts of network activity data, which is then used to identify an optimal network configuration design reflecting estimated demand. Use of dedicated hardware means that this approach is relatively expensive, incurring costs in hardware procurement, depreciation and maintenance, as well as significant training and operational costs. In this paper we outline the foundations of an efficient, QoS-aware network planning process, that, through re-use of networking accounting data, can be delivered at relatively low cost and with minimal impact on network performance. We contend that ISPs can easily reuse traffic data gathered by network accounting systems to generate sufficiently accurate estimations of network demand. In this paper we show that such reuse of accounting data to construct a QoS-aware demand matrix results acceptable relative error for large amounts of traffic. We also present a light-weight technique for estimation of QoS aware effective bandwidth coefficients, which in conjunction with the QoS-aware demand matrix provides the basis for an effective, low cost QoS-aware network planning process.

The paper is organised as follows. §2 disuses related work in the area of network accounting and network planning. §3 provides a description of our architecture, detailing the components and tools used to gather and analyse traffic data. We present our algorithm for calculating the demand matrix from accounting records in §4. §5 discusses how we plan to take QoS into consideration by estimating effective bandwidth coefficients. We validate our architecture in §6 through the use of a specified scenario and a prototype test bed. Finally we evaluate our results in §7 and conclude with future work in §8.

2. Related Work

The concept of network planning for QoS aware network optimisation has been put forward by Wu and Reeves [1]. They look at capacity planning in a DiffServ network, and focus on a network with two traffic classes, Expedited Forwarding (EF) and Best Effort. Their work focuses on developing an optimisation algorithm that jointly, selects a route in the network for each EF user demand (Origin - Destination) pair, and assigns a capacity value for each link within the network to minimise the total link cost, subject to the performance constraints of both EF and BE traffic classes. Our work intends on developing a network planning solution to take into consideration all DiffServ traffic classes.

The demand matrix has been associated with a wide range of network planning activities such as network design, traffic engineering and capacity planning [1]. The demand matrix has been shown to be an effective method of representing network wide demand on the network from edge to edge [2, 3, 1, 8]. There are a number of different approaches in calculating the demand matrix, the major concern being whether to calculate the demand matrix from direct measurement [3], or to use summarised sampling methods such as trajectory sampling [9]. The algorithm we propose is a centralised approach to calculating the demand matrix. This is primarily based on the work of Feldman et al [3]. Our algorithm varies as it is limited to the types of metering records used by accounting systems. For efficiency processing may also be distributed further out onto the edge nodes themselves [4].

The IETF have developed a number of network accounting architectures such as RADIUS [11], and DIAMETER [12]. These systems rely on the collection of metering information from the network, and forward this information to mediation points following a particular format. These systems are most commonly used for VoIP accounting and other such session based services. The IP Multimedia Subsystem has defined DIAMETER as its accounting protocol of choice [13]. Cisco have developed NetFlow [14] as their metering and accounting system. NetFlow is widely used in the industry for various operations such as IP network accounting and billing, user and application monitoring, network planning, security analysis and traffic engineering. The IETF have recognised this industry standard and have developed the IETF IP Flow Information Export (IPFIX) [1] architecture based on it. We have based our accounting system on the IPFIX architecture.

3. QoS-aware Network Planning Architecture

The architecture illustrated in Fig. 1. extends the traditional network accounting architecture to facilitate construction of QoS-aware demand matrices and for the calculation of QoS-aware effective bandwidth coefficient from collected packet traces.

All network accounting systems depend on metering information to account for service usage within their network domain. The network accounting systems capture summarised information from the network in the form of flow records, of the form depicted in below.

Src Address	Src Port	Dest Address	Dest Port	Protocol	TOS	Packets	Size	Start time	Active	Idle

A flow record represents a set of IP packets passing a network interface that possess common properties, such as the same source address, source port, destination address destination port and protocol. A flow record can usually be associated with unidirectional traffic of a particular application session, such as a VoIP call. An example of network accounting systems that use flow records are Cisco Netflow [14], and the IETF proposed standard IPFIX [1]. These records are used as a base for rating usage of traffic within the network. Once this traffic is rated, it can be associated with user sessions where the users can be charged and billed for service usage. We intend

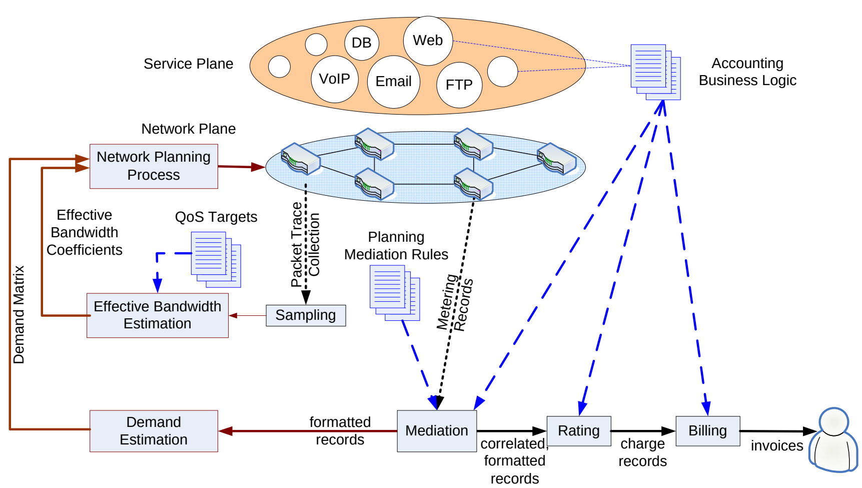


Fig.1 Network Accounting and Planning Architecture

to use the flow records collected by accounting systems to construct a view of the traffic demand on the network, other wise known as a demand matrix.

The demand matrix captures network wide demand from information collected at ingress and egress edge routers. We assume that all traffic entering at an ingress edge node, must exit at an egress edge node, i.e. no traffic is consumed within the core. If this is the case, we can build a picture of total traffic demand within the network between edge nodes. The demand matrix stores this information as a pair-wise edge-to-edge matrix. To create this matrix, we must have knowledge of where traffic exits the core network, and match this against its ingress. We wish to build a demand matrix for each particular traffic class on the network as to associate particular traffic class demand characteristics with QoS targets within the accounting system. This is achieved by adding an additional dimension to the demand matrix matching the *Type of Service* (TOS) field of metered traffic.

The TOS field identifies the QoS traffic class of a particular IP packet. This information is carried on into the flow records. The DiffServ QoS architecture uses the TOS field to assign DiffServ Code Points (DSCP) to IP packets to include them into particular traffic classes. QoS levels for these traffic classes can then be controlled within the network, giving the traffic class particular loss, delay, jitter, and throughput targets. By adding the TOS dimension our demand matrix becomes QoS-aware.

The *Mediation* component exports collected metering records in the required format to the *Demand Estimation* component. This process will construct a multi-dimensional demand matrix of all accounted traffic passing through the network from edge to edge and per traffic class. Our observation is that accounting systems record a considerable amount of network activity, while at the same time summarising this information to reduce processing and storage. We contend that using existing metered information to estimate the demand matrix is less expensive than calculating the demand matrix through the use of dedicated hardware using direct measurement methods. Of course, this data may not provide an accurate measurement of network demand, however, as shown below, the relative error introduced is unlikely to impact significantly on the efficacy of the planning process.

A relatively small number of packet traces are taken from the network and analysed to calculate appropriate bandwidth thresholds known as effective bandwidth to be reserved in order for this traffic to maintain defined QoS targets. The process of packet trace collection is a very light weight approach to sampling network activity over a short interval of time, which has little effect on network resources.

Once the demand matrix and effective bandwidth coefficients are estimated, they are used as input to the *Network Planning* component, which will use the collected and analysed information to develop an optimised network configuration that will ensure imposed network QoS targets are maintained.

4. Demand Matrix Estimation Process

Accounting systems mediate metering records to associate service usage with users. Similar to this approach we need to associate the metering records with edge to edge network demand. We outline our algorithm to achieve this in Fig. 3. The algorithm calculates the demand on the network a particular flow has, per interval. This is necessary as the demand matrix looks at total network demand per interval of time, e.g. 10 min intervals.

Fig. 2 depicts an algorithm of estimating network demand from flow records per interval. Each flow record will have a start time (t^f_{start}) and an end time (t^f_{end}). The flows rate r^f can be calculated from the flow size, stored in the flow record, divided by the flow duration. The diagram shows 4 cases the algorithm captures. The objective of the algorithm is to sum up all demand of all flows that lie within a particular time period $\{t_n\ ,\ t_{n+1}\}$. Case 1 captures demand of flows that end within the time period. Case 2 captures demand of flows that start within the time period. Case 3 captures demand of flows that start and end within the time period, and finally Case 4 captures

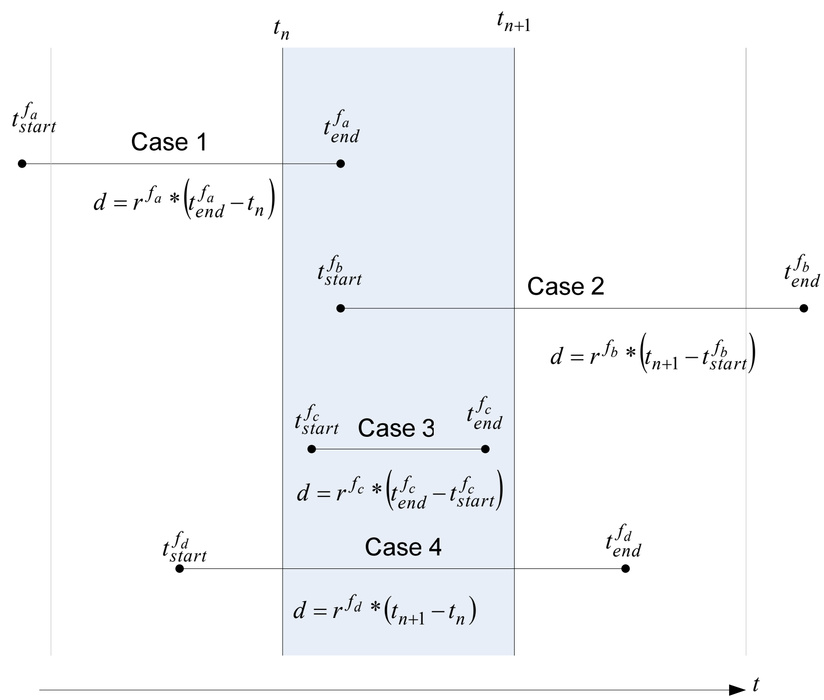


Fig. 2 Estimation of flow demand per interval of time

```
Input:(EdgeRouters, interval, TotalTime  )
Output:(DemandMatrix[IngressRouter, EgressRouter, TOS] )
For each ingressRouter in EdgeRouters
   For each meteringDevice in edgeRouter
      For each sourceNode of meteringDevice
         For each timeInterval in TotalTime
            For each flow of meteringDevice
               If flow_src_ipaddr == sourceNode
                  Demand =
                        calculateDemand(timeInterval,
                        interval, flow)
                  egressRouter =
                        findEgressRouter(flow_dest_ipaddr )
                  DemandMatrix[ingressRouter, egressRouter,
                        flow_TOS ] += Demand
 Return DemandMatrix
```

Fig. 3 Demand Matrix Estimation Algorithm

demand of flows that are active through the whole time period. This algorithm makes the assumption that packet arrival time within the flow is of a uniform distribution. By taking this assumption the proportion of demand can be calculated by multiplying the flow rate by the duration of time the flow exists within the current interval. This will lead to some inaccuracy in the final value, as throughout the flow's duration, packet distribution is not normally uniform. To calculate the complete network wide demand matrix all metering records collected from on all metering devices on all ingress edge routers are processed.

The algorithm expressed in Fig.3 has five nested for loops, looping through each ingress router, each metering device, and each source node hanging off that metering device, each interval in time, and each flow record within that current metering device. The algorithm matches each flow record to a source node, and estimates the flow's demand within the current interval. The algorithm then matches the destination address of the flow record to a particular egress edge node. This mapping allows us to identify where the traffic is exiting the network.

The *findEgressRouter* function in Fig. 3b is used to return the egress router node corresponding to where the flow exits the core network. This function is a simple static table lookup at the moment but can be extended to retrieve this value dynamically through BGP table lookups. Once the egress edge node is found, an entry is added to the demand matrix. Once all records have been processed the demand matrix is returned.

5. Taking QoS requirements into account

The network planning process requires a view of demand on the network. We propose estimating network demand by the generation of a demand matrix from accounting records. We wish to develop a network planning scheme that will provide improved QoS guarantees. To take QoS targets into account per traffic class, we propose a light weight method of estimating the effective bandwidth of a traffic class, between edge nodes.

There are a number of definitions of what effective bandwidth is, for example see [4, 5]. From the different methods of estimating the effective bandwidth, practically all of them are based on building a traffic model. The building of an accurate traffic model for a bursty traffic source is quite a challenging task. In particular, it is quite difficult to take into account the different activity levels of a traffic source for different time scales ranging from milliseconds to minutes and hours.

The effective bandwidth of a traffic source is a minimal link rate which can guarantee certain specified QoS targets. Effective bandwidth can be defined for different types of QoS targets including delay, loss or both delay and loss targets together. In this paper we are interested in QoS delay targets only. A QoS delay target specifies the maximum delay experienced on the network and the proportion of traffic which is allowed exceed this maximum delay.

A typical example of a QoS delay target is (50ms, 0.001) which means that only 0.1% of traffic is allowed to be delayed more than 50 ms. As effective bandwidth depends on the QoS target; for different QoS targets, effective bandwidth could be different.

Our approach of estimating effective bandwidth follows a more empirical method and fits well to our main objective related to QoS aware network planning. A brief description of the algorithm we wish to implement is as follows. Suppose the QoS delay target ($delay_{max}$, p_{delay}) is fixed and includes $delay_{max}$ the maximum delay and p_{delay} the percentage of traffic which can exhibit delay more than $delay_{max}$. We define effective bandwidth R_{eff} of a traffic source for delay QoS target ($delay_{max}$, p_{delay}) as a minimal link rate such that if we simulate a FIFO queue (with unlimited buffer) the percentage of traffic which will exhibit delay more than $delay_{max}$ will be less than p_{delay}. We will assume that initially the queue is empty.

To estimate the effective bandwidth of a particular traffic source on the network, we take a recorded packet trace of that source. The algorithm we define for estimating the effective bandwidth of a recorded packet trace is as follows. The algorithm is based on the following observations. Suppose we simulate a FIFO queue with the same traffic source for different queue rates $R_1 > R_2$ and estimate the percentages p_1 and p_2 of traffic delayed more than $delay_{max}$ for different rates respectively, then $p_1 \leq p_2$. This means that the percentage of traffic delayed more than $delay_{max}$ is a monotonically decreasing function of the queue rate. Using this observation it is straight forward to design an algorithm for a recorded packet trace to find the minimal value of a queue rate such that the percentage of traffic delayed more than $delay_{max}$ is less than p_{delay}.

$$k_i = \frac{R_{eff,i}}{mean_i}$$

We assume that the QoS delay target ($delay_{max}$, p_{delay}) is fixed per traffic class. We take a large number of recorded traffic traces of more or less the same duration T_{max}. The choice of T_{max} is important. If T_{max} is too large or too small, the estimated ratio of effective bandwidth to the mean rate will be underestimated. Typically T_{max} is chosen between 1 minute and 1 hour, e.g. 10 minutes. Suppose we have N traffic traces. For the i^{th} traffic trace we estimate both $R_{eff,i}$ and $mean_i$ and calculate

We note that the effective bandwidth is always larger or equal to the mean rate. So for all i, $k_i \geq 1$. We now consider a set of N effective bandwidth coefficients $\{k_1, \ldots, k_N\}$. First we exclude any k_i with too small a mean rate using some appropriate threshold value. Second we calculate K_{95} the 95th percentile. The effective bandwidth coefficients K_{95} is used for our purposes.

6. Experimental Evaluation

Fig 4. illustrates an ISP providing connectivity and services to a number of customer groups. A customer group defines a set of service interactions with a number of services offered by the ISP to fulfil a particular business process. The ISP offers various levels of QoS guarantees to the customers depending on the type of service the customer requests and the amount of revenue generated by that service interaction. The ISP therefore has a set of QoS targets it must maintain within its network in order to generate maximum revenue.

The ISP has a number of application servers distributed through the network offering services to the customers. The customer is located at a number of customer group LANs distributed throughout the network. The ISP has an accounting system in place to meter service usage. Metering information is collected at key points throughout the network through the use of strategically positioned metering devices.
This metering information is then used for rating and billing purposes by the service providers accounting system. The initial step of our proposed network planning

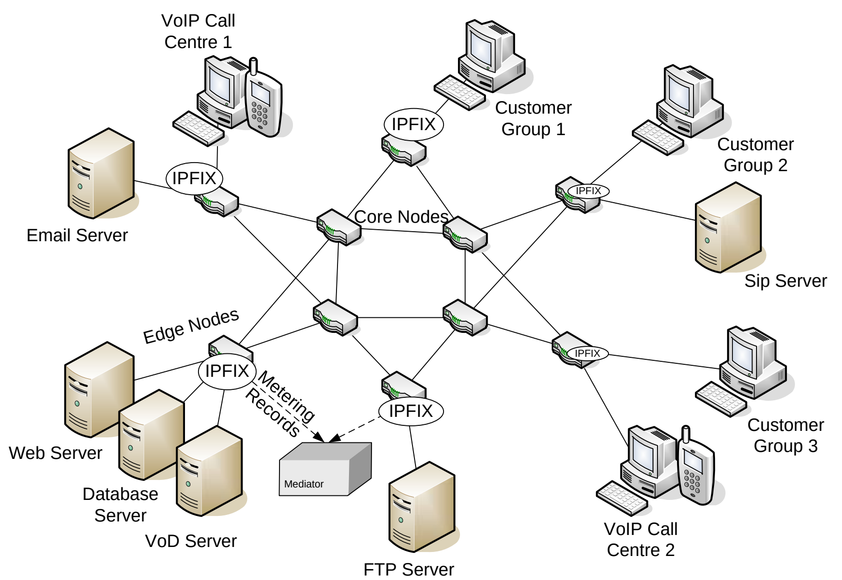


Fig. 4 Scenario Topology

Table. 1 Traffic Pattern Setup

Service	DSCP	Pattern 1	Pattern 2	Pattern 3
SIP VoIP	EF	5 users	15 users	30 users
FTP	AF21	5 users	5 users	5 users
Email	AF31	10 users	10 users	10 users
Database	AF41	5 users	5 users	5 users

Table. 2 Service Characteristics

Service	Usage pattern
SIP VoIP	Silence length is exponentially distributed with a mean of 0.65s Talk spurt length is exponentially distributed with a mean of 0.352s Encoding rate is 8 Kbps
FTP	Inter request time is exponentially distributed with a mean of 720s File size is a constant 5000 bytes
Email	Send / receive interval is exponentially distributed with a mean of 720s Email size is a constant 3000 bytes
Database	Transaction interval is exponentially distributed with a mean of 12s

process utilises this accounting information to estimate the demand imposed on the network by service interactions.

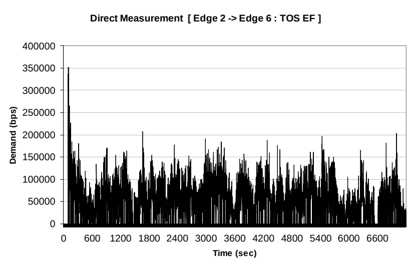
To test and validate our proposed algorithms in Fig. 3 we have implemented a use case based on the scenario with a prototype network topology. We use OPNET as our network simulation environment. We compare our approach of calculating the demand matrix to direct measurements taken, and show that our approach produces acceptable relative error. The network topology has four core routers, and five edge routers, each connected by 10Mbps Ethernet. All IP traffic is generated at the edge of the network, no IP traffic is consumed at the core of the network.

There are three customer groups each containing 25, 35 and 50 customers respectively. There are four services the service provider offers to the customers on the network. They are email, FTP, database and VoIP (voice over IP). These are located on application servers distributed around the edge of the network. An IPFIX device has been modelled in OPNET and attached to each ingress interface of the edge nodes. All traffic entering the network is available for both direct measurement and accounting based metering. Each of the four services is set up to generate a particular traffic pattern within the network for customer group 2, see Table. 1. Customer group 1 and 3 will have set traffic patterns for all three simulations. For each simulation Customer group 2 will follow a different traffic pattern; each traffic pattern will have an increased number of VoIP users on the network, thus increasing the amount of EF traffic generated across two particular edge nodes. Each customer within a group is set up to interact with the offered services following a particular service usage pattern, outlined in Table 2.

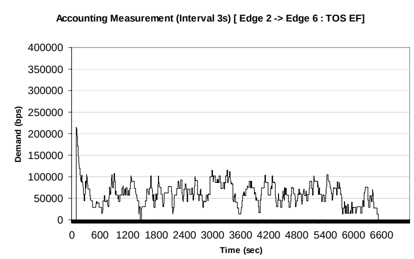
7. Experimental Results

We ran the OPNET simulation over a 2 hour simulation period, collecting direct measurements and accounting records from all ingress interfaces of the edge nodes. Fig. 5 shows demand across edge 2 to edge 6 for traffic class EF. Fig.5a. shows the direct measurement of demand calculated from direct measurements. This graph has the highest resolution of accuracy. Fig.5b-d show the same edge to edge demand for the same traffic class, but are generated from the accounting records. In each case the interval over which the demand is calculated increases from 3 to 30, to 100 seconds. This means the demand values calculate for the demand matrix are over these three interval steps. The figures above show slight loss in accuracy by reducing the size of the sampling interval.

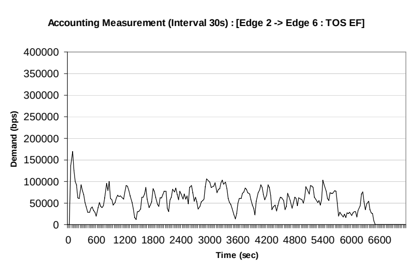
We calculate average demand over 10 min intervals for the three traffic patterns outlined in Fig. 6. By this we mean from the values held in the demand matrix, we calculate demand over 10 minute intervals, and compare these values to direct measurements. Traffic pattern 1 has the lowest demand over the simulation duration between edge 2 and edge 6 for traffic class EF. Traffic pattern 2 has a slightly larger



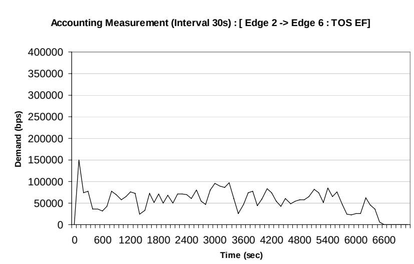
(a). Direct Measurement



(b). Accounting : Sample Interval 3s



(c). Accounting : Sample Interval 30s



(d). Accounting : Sample Interval 100s

Fig. 5 Direct measured demand vs. accounting based demand

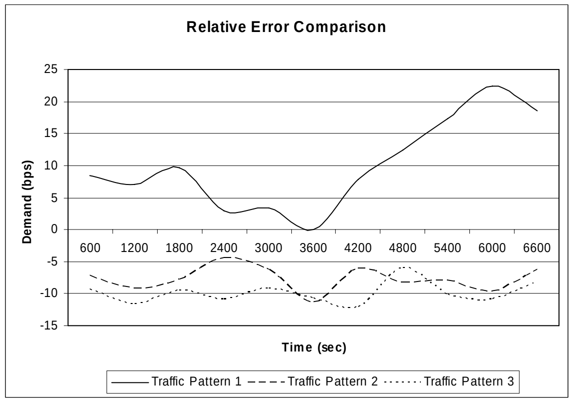


Fig. 6 Relative error across all traffic Patterns

demand across the edges and traffic pattern 3 has the largest demand across the two edges.

We take three traffic patterns and run them for a period of 2 hours. We calculate relative error in calculating demand based on accounting records in comparison to their associated direct measurements over 10 minute intervals. From these experiments we observe that our approach estimates demand with relative error of approximately 10 %. This can be tied to the fact the our demand estimation algorithm is based on analysing accounting records, which are in turn a summary of network traffic. As network planning is predominantly based on estimation of current network demands, a high level of accuracy is generally not required as future traffic demands are dependent on human usage trends, which are in themselves unpredictable. Therefore margin of error in demand estimation is quite acceptable for the purpose of QoS aware network planning, of which the demand matrix is a vital part of.

8. Conclusions and Future Work

We proposed a method of estimating network demand from pre-existing flow records used by network accounting systems, and a method of calculating QoS related effective bandwidth coefficients. These coefficients tell us how much bandwidth is required per traffic class for services to meet QoS targets. We can use these coefficients with the QoS aware demand matrix to develop a network planning solution specific to the QoS targets outlined by the network operator and between the service provider and customer. Future work will focus on specification and evaluation of a complete network planning process based on the network demand and QoS-aware effective bandwidth coefficient estimation techniques outlined here. We also intend to investigate incorporation of business level input regarding future service demand trends.

Acknowledgements

The authors would like to take this opportunity to thank the anonymous referees for their useful comments and feedback on the paper. This work has received partial support from the following: M-Zones is funded under the HEA PRTLI, Science Foundation Ireland under the Autonomic Management of Communications Networks and Services programme (PI Cluster Award 04/IN3/I404C).

References

1. K. Wu and D. S. Reeves, "Capacity Planning of DiffServ Networks with Best-Effort and Expedited Forwarding Traffic," Telecommunications Systems, vol. 25, no. 3/4, pp. 193-207, (2004).
2. G. Sadasivan, N. Brownlee, B. Claise, and J. Quittek, "Architecture for IP Flow Information Export (Internet-Draft)," draft-ietf-ipfix-architecture-11.txt, http://www.ietf.org/html.charters/ipfix-charter.html (2005).
3. A. Feldman, A. Greenberg, C. Lund, N. Reingold, J. Rexford, and F. True, "Deriving traffic demands for operational IP networks: methodology and experience," IEEE/ACM Transactions on Networking (ToN), vol. 9, no. 3, pp. 265-280, (2001).
4. K. Papagiannaki, N. Tatf, and A. Lakhina, "A Distributed Approach to Measure Traffic Matrices," Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement (2004).
5. F. Kelly, "Notes on effective bandwidth," in Royal Statistical Society Lecture Notes Series, 4. Oxford University Press, pp. 141-168,(1996).
6. J. Y. Hui, "Resource allocation for broadband networks," IEEE Journal on Selected Areas in Communications, vol. 6, pp. 1598-1608, (1988).
7. D. D. Botvich and N. Duffield, "Large deviations, the shape of the loss curve, and economies of scale in large multiplexers," Queueing Systems, vol. 20, pp. 293-320, (1995).
8. A. Medina, C. Fraleigh, N. Tatf, S. Bhattacharya, and C. Doit, "A Taxonomy of IP traffic matrices," Proc. International Society for Optical Engineering (SPIE) Vol. 4868, p. 200-213, (2002).
9. A. Soule, A. Lakhina, N. Tatf, K. Papagiannaki, K. Salamatian, A. Nucci, M. Crovella, and C. Doit, "Traffic matrices: balancing measurements, inference and modeling,", Joint International Conference on Measurement and Modeling of Computer Systems ACM Press New York, NY, USA, pp. 362-373, (2005).
10. N. Duffield and M. Grossglauser, "Trajectory sampling for direct traffic observation," New York, NY, USA: ACM Press, pp. 271-282,(2001).
11. C. Rigney, "RADIUS Accounting," in IETF RFC 2139 Internet Engineering Task Force, www.ietf.org/rfc/rfc2139.txt, (1997).
12. Calhoun P., Loughney J., Guttman E., Zorn G., and Arkko J., "Diameter Base Protocol," in RFC 3588 Internet Engineering Task Force, www.ietf.org/rfc/rfc3588.txt, (2003).
13. 3GPP2, "IP Multimedia Subsystem - Accounting Information Fows and Protocols," in All IP core network multimedia domain, 3GPP2 X.S0013-008-0, (2005).
14. Cisco Systems, “Introduction to Cisco IOS Netflow – A Technical Overview”, last accessed: February 2006, http://www.cisco.com/en/US/products/ps6601/, last visited 01/08/2006.