IEEE ICC WORKSHOP FUTURENET V, Ottawa, Canada, June 2012

Traffic Provisioning in a Future Internet

T.H. Szymanski and S. Behdin
McMaster University, Hamilton, Canada
teds@mcmaster.ca

***Abstract*—An adaptive traffic estimation algorithm for an *Autonomic Future Internet* which can provide improved throughput, energy-efficiency and QoS guarantees is proposed. The theory for a *Future Internet* which supports multiple service classes, i.e., the traditional *Best-Effort* (BE) class and a new *Essentially-Perfect-QoS* (QoS) class, has recently been proposed. In this *Future Internet*, the routers give preference to the QoS traffic class. All smoothened end-to-end traffic flows in the QoS class will never experience congestion and can achieve *Essentially Perfect* link-utilizations and end-to-end QoS guarantees, with significantly improved energy-efficiencies. Each *Future Internet* router must provision bandwidth for the QoS traffic class, and schedule this class with 100% throughput efficiency and strict QoS guarantees, using a recently-proposed mathematical scheduling algorithm. In this paper, adaptive traffic estimation algorithms which allow each *Future Internet* router to estimate its future QoS traffic demands and provision bandwidth for the QoS demands in anticipation of their arrival are proposed. An *Autonomic Controller* (AC) in each router maintains a history of its QoS and *Best-Effort* traffic demands over a long time horizon. Several variations of *Autoregressive Integrated Moving Average* (ARIMA) filters are used to estimate the future traffic demands. The AC can then provision resources for the QoS demands in anticipation of their arrival. To test the algorithms, real traffic measurements taken every 15 minutes over 4 months for a European backbone network are used. The estimates are shown to be very accurate, provisioning bandwidth for the QoS class with success rates between 93% ... 99%. An *Autonomic Controller* (AC) in each router can also be used to automate the bandwidth provisioning process for existing *Differentiated-Services* traffic classes in existing *Best-Effort* Internet routers.**

***Index Terms*—Future Internet, autonomic, controller, QoS, traffic estimation, ARIMA filters, energy efficiency**

I. INTRODUCTION

The *Best-Effort* Internet network is a universal platform for providing new services. However, the *Best-Effort* Internet faces challenges including a reliance on significant over-provisioning to achieve relatively-poor QoS guarantees [1], which leads to poor resource-utilization and poor energy-efficiency. The inefficiencies of the *Best-Effort* Internet contribute to green-house gasses and global warming, and are estimated to cost hundreds of millions of dollars in excess energy costs annually. In addition, the current practice of over-provisioning the *Best-Effort* Internet to achieve relatively poor performance guarantees leads to excess capital expenditures, to build an infrastructure which remains largely under-utilized on average. To address these problems, governments around the world are exploring the *'Future Internet Architectures'*, and are open to both *evolutionary* and *revolutionary* changes to the *Best-Effort* Internet.

In this paper, algorithms to estimate and provision traffic demands in a recently-proposed *Future Internet* network are proposed. In this paper, estimated traffic demands are used to provision resources for a new QoS traffic class in anticipation of their arrival. A software-based *Autonomic Controller* (AC) can be introduced into each router. The AC in each router performs the traffic estimation and resource allocation processes, a step towards a fully *Autonomic Future Internet* or more general *Autonomic Computing* system [5].

The *Future Internet* model we consider supports multiple service classes, i.e., the existing *'Best-Effort'* (BE) class, along with a new and highly-efficient *Essentially-Perfect* (QoS) class [16,17]. The QoS class can contain high-bandwidth aggregated video traffic from cloud-based servers such as YouTube, Netflix and iTunes, or traffic for cloud-based computing systems. It has been estimated that video-traffic will soon account for the majority of all Internet traffic. Traffic in the new QoS class is highly-efficient and can achieve significantly better link-efficiencies, energy-efficiencies and QoS guarantees compared to today's *Best-Effort* Internet [16,17].

The new QoS traffic class can also lead to considerable capital-cost savings by removing the need to significantly over-provision the *Best-Effort* Internet. Today's *Best-Effort* routers typically operate at light loads (i.e., 33%) to provide statistically low delays, representing a significant loss of capacity, and a significant capital cost to pay for over-provisioning capacity [1]. By allowing the network to achieve mathematically-provable and very low delays even at 100% loads, the capacity of an entire backbone network can be increased representing a considerable capital-cost savings, potentially measured in the billions of dollars per year. By significantly increasing the capacity of the Internet infrastructure, the energy-efficiency of the entire network also improves significantly.

To achieve strict and mathematically-provable QoS guarantees, each *Future Internet* router must schedule the QoS traffic flows to achieve 100% efficiency and strict QoS guarantees, using a recently-proposed mathematical scheduling algorithm. Achieving 100% throughput and strict QoS guarantees in one input-queued switch or router without over-provisioning is a well-known and difficult problem [4,16,18], which is summarized in section 2.

The *Future-Internet* network described in [16,17] can use a reservation-based model such as MPLS, where RSVP is used to reserve resources for each end-to-end QoS traffic flow, to provide *Essentially-Perfect* end-to-end QoS guarantees. Our traffic estimation algorithms provide estimates of future QoS demands, which can be provisioned in anticipation of their arrival. However, the same algorithms can be used to automate the resource allocation process for the *Differentiated Services* (DiffServ) model in existing *Best-Effort* Internet routers. DiffServ does not use end-to-end reservations. In a DiffServ network, each router can allocate bandwidth for each DiffServ traffic class on its outgoing links. Currently, this DiffServ allocation process is not automated and is performed manually in each router (if it is used at all). An *Autonomic Controller* in each router can also use the proposed traffic estimation algorithms to provision bandwidth for DiffServ traffic classes in each router in the existing *Best Effort* Internet network.

The AC in each router maintains a history of its traffic demands over a long time horizon. The traffic rates between Source-Destination (SD) pairs can be low-pass filtered to reduce transients. Differences of the filtered rates can be used to estimate the future rates. To provision resources for a end-to-end QoS traffic flow, the source node can perform a source-based routing algorithm. The routing algorithm will compute a desired end-to-end path between the SD pair, which optimizes a performance metric of the network, i.e., minimum cost, minimum delay, maximum reliability, etc. In our application, any good source-based routing algorithm can be used.

In a reservation-based network model, the AC can perform source-based routing, and reserve resources for the end-to-end traffic flows using a signalling protocol such as RSVP. In a non-reservation-based network model (i.e., a DiffServ network), the AC can use allocate bandwidth for each DiffServ traffic class along the outgoing edges of the router. Every DiffServ router in the network can perform the same autonomic tasks, thereby provisioning bandwidth for the DiffServ classes throughout the network, in each provisioning interval.

Several techniques have been proposed to estimate traffic demands in a network. Reference [7] has summarized several different traffic modelling techniques, including discrete-event simulation, renewal-process models, markov-modulated models, fluid models, linear autoregressive models and self-similar models. Several papers have proposed the use of Kalman filters to estimate traffic demands [14,15]. Kalman filters are based on *Linear Dynamic Systems* with a discretize-time model. They model a hidden *Markov chain* built upon linear operators where measurement perturbations drawn from a Guassian distribution with zero-mean. Kalman filters and Markov chain models select an estimate for the next-state based upon the current estimated state and the current measurements; no long-term history is modelled. Unfortunately, it is well-known that real Internet traffic is usually self-similar, with a very high burstiness. Self-similar traffic is also characterized as having large variances and significant autocorrelations over long time horizons [3]. The characteristics of real Internet traffic suggests that other models which can consider longer time horizons may be better for modelling Internet traffic.

Our traffic models can be classified as variations of the *Autoregressive Integrated Moving Average* (ARIMA) filters.

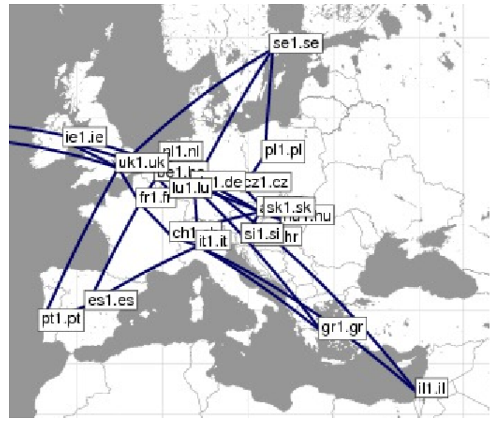

Fig. 1. The 'Geant' European backbone network [2].

A general ARIMA model has the form

$$X_n = a_0 + \sum_{r=1}^{p} a_r X_{n-r} + \sum_{r=1}^{q} b_r \epsilon_{n-r}, n > 0 \qquad (1)$$

where the $X_0, ..., X_{p-1}$ are random variables, the a_r and b_r are real constants (coefficients), and the ϵ_r are error terms, typically zero-mean IID random-variables which are independent of the X_n. In our traffic models, several ARIMA-type filters are created, each operating on a different time-scale, i.e., quarter-hours, hours, days and weeks. Each filter has a weight which reflects its accuracy in predicting the future traffic demand. The weights change *adaptively* as time progresses, so that the most accurate filter has the largest weight when forming the next traffic estimate.

To test the estimation algorithms, real inter-city traffic measurements taken every 15 minutes over a period of 4 month for the European 'Geant' backbone network shown in Fig. 1 are used [2]. The traffic estimation algorithms are shown to be very accurate, provisioning sufficient bandwidth with success rates typically between 93% - 99%.

In the proposed *Future Internet*, resources reserved for the QoS class can be used by the BE class if the QoS traffic demand does not materialize. Similarly, if unanticipated QoS traffic demands arrive, then they can be provisioned in real-time (as is currently done in MPLS or ATM networks). The AC relieves the load of provisioning all demands in real-time.

Section 2 provides a brief review. Section 3 summarizes the *Future Internet* model we assume. Section 4 presents the proposed estimation algorithms. Section 5 concludes the paper.

II. Review

A. *QoS Scheduling*

It is well known that a *Maximum Weight Matching (MWM)* algorithm can be used to schedule the transmission of packets through an Input-Queued switch or router and achieve essentially 100% throughput [4]. However, the MWM algorithm requires $O(N^3)$ computation per time-slot, rendering it intractable for Internet routers. The problem of scheduling multiple competing traffic flows within a single IQ switch, to achieve 100% throughput and strict QoS guarantees, is difficult. For example, researchers at Bell Labs. have shown that scheduling multiple competing traffic flows within a single IQ switch to minimize jitter is NP-HARD [18]. Any practical solution to this QoS scheduling problem can be applied to the current Internet IP and MPLS networks, which currently use heuristic *Best-Effort* schedulers. Recently, a fast polynomial time solution to the QoS scheduling problem has been presented (see [16]). The algorithm can be applied to yield the proposed Future Internet network, with improved resource-utilization, energy-efficiency and QoS guarantees [16,17].

B. Review of Traffic Models, Estimation and Provisioning

Leland et al. have shown that real Internet traffic is often self-similar, exhibiting burstiness over several time-scales, which makes traffic modelling and provisioning difficult [3]. Gunnar et al have described a Global Crossings backbone IP network, where traffic flows are provisioned between cities in each provisioning interval [6]. Barakat et al have presented a wavelet model to model Internet traffic [8]. Freleigh et al have reported traffic measurements on a Sprint IP backbone network [9,10]. Reference [10] reports a technique to estimate traffic demands and provision end-to-end traffic flows (using RSVP), to reduce the delay per flow. Reference [13] describes ARIMA filters used to estimate long term NSF backbone traffic. References [14,15] present traffic models based on Kalman filters.

III. The Future Internet Network Model

A backbone Internet network as shown in Fig. 1 can be represented as a directed graph $G(V,E)$, where V is the set of fixed routers, and E is the set of directed edges. The proposed *Future Internet* network supports multiple service classes, i.e., the usual *Best-Effort* (BE) class along with a new *Essentially-Perfect* QoS class [16,17]. All legacy *Best-Effort* Internet applications developed over the last 40 years continue to run over the proposed *Future Internet* network, while new applications can also be written to exploit the new and highly-efficient QoS class.

The traffic demands in an Internet backbone network can be characterized by several types of traffic rate matrices. Let T^{QoS} and $T^{BE} \in R^{N \times N}$ be *global traffic demand matrices*. Element $T^{QoS}(i,j)$ denotes the bandwidth requirement for QoS traffic between the pair of cities (or source-destination routers) (i,j). Similarly, element $T^{BE}(i,j)$ denotes the bandwidth requirement for Best-Effort traffic between the pair of cities (or source-destination routers) (i,j).

The routing algorithm Γ can be centralized or distributed; it can use single-path or multi-path routing. In the Global Crossing MPLS-TE network described in [6], the routing is performed by the source routers using a constraint-based routing algorithm, and RSVP was used to reserve resources for each *Label-Switched-Path* (LSP) according to the routing. The proposed Future Internet can use the same routing methodology. Each traffic flow to be routed specifies a QoS bandwidth requirement. When the flow is routed, RSVP can be used to reserve the QoS bandwidth in each router along the end-to-end path for the flow. Each router k will update its local *router traffic rate matrix* D_k^{QoS} in response to the RSVP messages. Each router can schedule its QoS traffic flows to achieve hard QoS guarantees, using the algorithms described in [16].

The global traffic demand matrix T^{QoS} will evolve over time. The *'Autonomic-Controller'* in each router will maintain the recent history of these matrices on a periodic basis. Define a *provisioning interval* as an interval of time in which traffic demand matrix is measured and recorded, i.e., every 15 minutes. The memory requirements for recording the history of these matrices is relatively small. Each matrix entry requires 2 bytes to provide a reasonable resolution on the bandwidth requirement. The history of one 50x50 matrix over 1 year at 15 minute intervals requires only 175 Mbytes (without compression), which is negligible compared to the memory in an iPod music player.

A. Autonomic Support for DiffServ Traffic Classes

The DiffServ model provides 3 basic traffic classes in order of decreasing priority, the *Expedited-Forwarding* class, the *Assured-Forwarding* class, and the *Default* (or Best-Effort) class. Existing *Best-Effort DiffServ* routers give preferential treatment to the traffic classes with higher priority, however they cannot provide any hard QoS guarantees due to their *Best-Effort* nature. Currently, each *Best-Effort DiffServ* router must be manually programmed to provide bandwidth for a Diff-Serv traffic class, and each manufacturer (Cisco, Juniper Networks, Alcatel, etc) has a different methodology to configure their Diff-Serv routers. As a result, many network operators do not use Diff-Serv. Many networks currently use a single *Best-Effort* traffic class and rely upon the *significant over-provisioning of bandwidth* to achieve lower queueing delays and improved performance [1].

In another application of our traffic models, bandwidth for DiffServ traffic classes can be estimated and dynamically provisioned by the AC in each router. The AC can provision bandwidth on each outgoing link of a router for each DiffServ traffic class in each provisioning interval, thereby realizing an *Autonomic DiffServ Control Plane*.

IV. Traffic Estimation

The following notation is used to identify QoS traffic demands. Let $T(i,j)^t$ denote the QoS traffic demand between SD pair (i,j) at time t, where t represents a multiple of 15 minutes, starting from time 0, i.e., $T(i,j)^3$ denotes the traffic demand between SD pair (i,j) at time 45 minutes relative to time=0.

Let $\bar{T}(i,j)^t$ denote the low-pass filtered QoS traffic demand between SD pair (i,j) at time t. Let $\hat{T}(i,j)^t$ denote the estimated QoS traffic demand between SD pair (i,j) at time t, i.e., the *estimated bandwidth*. Let $\dot{T}(i,j)^t$ denote the amount of QoS traffic provisioned between SD pair (i,j) at time t, i.e., the *provisioning bandwidth*.

Fig. 2 illustrates real measured traffic over the Geant European backbone network (http://sndlib.zib.de/home.action). In

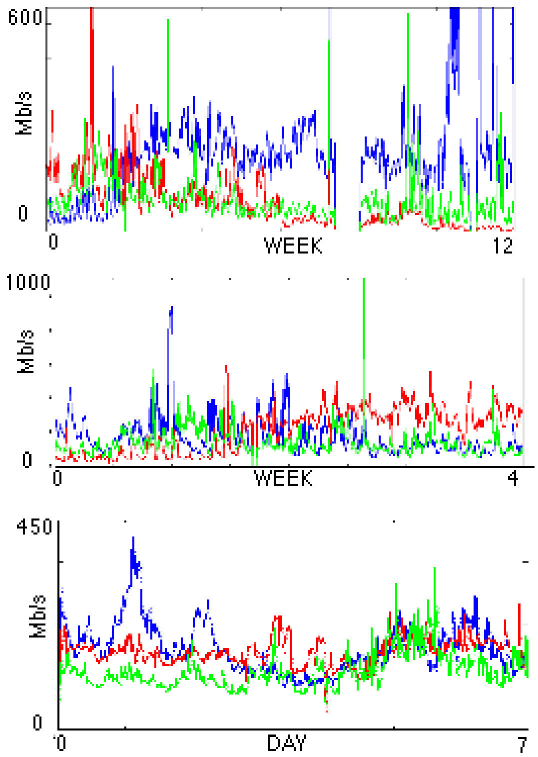

Fig. 2. Geant backbone traffic viewed at 3 different time-scales.

Fig. 2, the traffic intensity for the most intense 3 flows leaving one particular node are plotted on 3 different time scales, 12 weeks, 4 weeks, and 1 day. The traffic 'looks similar' over several time scales, with relatively bursty behavior on all time scales, suggesting some self-similarity characteristics.

The autocorrelation functions of all traffic leaving a node over a 2-week period are shown in Fig. 3, for 3 different nodes. There are significant long-term correlations in the traffic demands. For example, the traffic demands in each 15-minute interval are heavily correlated with previous 15-minute intervals, and with the demands in the same interval in the previous hours, days and weeks.

Define a window function $v = w(t_s, t_f, s)$ which operates on the vector $[T(i,j)^t]$ for $t = 1 : \infty$ and returns a vector v. The parameter t_s denotes a start time, t_f denotes an end-time. The vector v consists of the sequence of elements of $T(i,j)^t$ for $t = 1 : \infty$. where v(1) = the value of T(i,j) at the smallest time t $\geq t_s$, where v(k) = the value of T(i,j) at the largest t time t $\leq t_f$, and where successive elements of v are separated by time= s (the 'stride') in the vector $T(i,j)^t$ for $t = 1 : \infty$.

A. Traffic Estimation over a 15-Minute Time-Horizon

This estimation algorithm computes a single estimate for the next traffic demand value, based upon the current traffic demand value, the difference of the filtered traffic values at time t, and the variance of the traffic demands over a short time window. This model is based upon an *Autoregressive Integrated Moving Average* (ARIMA) filter, where the next state is a function of the current state and an estimated step based on the difference of the filtered traffic demands.

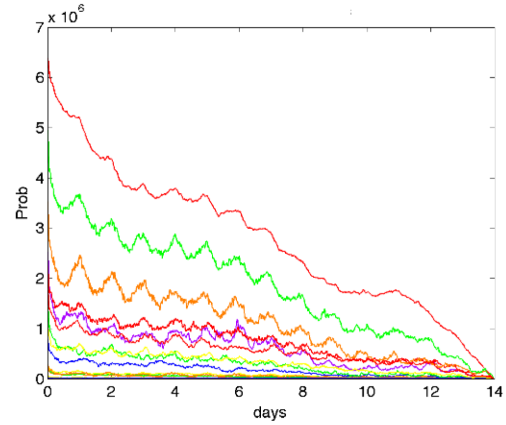

Fig. 3. Autocorrelations for several traffic flows leaving one node.

A short-term difference over the last 15 minute interval is computed as follows, based on the low-pass filtered traffic demands:

$$\Delta_q = \bar{T}(i,j)^t - \bar{T}(i,j)^{t-1} \tag{2}$$

An estimate for the traffic at time $t+1$ is given by

$$\hat{T}(i,j)_q^{t+1} = T(i,j)^t + \Delta_q \tag{3}$$

which can be rewritten in the form of Eq. (1), making explicit the low-pass filtering:

$$\hat{T}(i,j)_q^{t+1} = T(i,j)^t + \sum_{\tau=0}^{3} z_1(\tau)T(i,j)^t - \sum_{\tau=1}^{4} z_1(\tau)T(i,j)^{t-1} \tag{4}$$

for the low-pass filter parameters z_1. (Several types of low pass filters can be used. We used a simple moving-average filter, with four equal weights.)

A node can use this estimate to provision QoS bandwidth between cities in a backbone network, in anticipation of the demand in the next time interval. A variance σ_q^2 over a short-term window (i.e., 2 hours in the same day) can be computed as follows:

$$\sigma_q^2 = Var(w(t-8, t, 1) \tag{5}$$

The amount of bandwidth to provision between a pair of cities can include a constant k_q times the standard deviation. The last term determines how much 'excess bandwidth' is included in the estimated traffic demand to provision between cities, to accommodate for transients in the demand, where k_q is a constant. The *provisioning bandwidth* estimate is given by:

$$\dot{T}(i,j)^{t+1} = T(i,j)^t + \Delta_q + k_q\sigma_q \tag{6}$$

The results are shown in Fig. 4. The real traffic demands are shown in blue, while the estimated traffic demands are shown in red, for three (s,d) pairs. The curves are essentially super-imposed, indicating that the estimate is very accurate.

Several metrics can be used to evaluate the performance of the traffic estimate. The goal is to provision QoS bandwidth between a pair of cities to satisfy the demand in the next provisioning interval. The 'mean-satisfied-BW' α is defined as the mean value of the lower of the provisioning bandwidth estimate and the real bandwidth,

$$\alpha = (1/t) \sum_{\tau=0}^{t} min(\hat{T}(i,j)^{\tau}, T(i,j)^{\tau}) \tag{7}$$

The 'success-rate' β of the provisioning bandwidth $\dot{T}(i,j)$ is defined as the mean satisfied bandwidth divided by the mean bandwidth demand,

$$\beta = (1/t) \sum_{\tau=0}^{t} min(\hat{T}(i,j)^{\tau}, T(i,j)^{\tau})/T(i,j)^{\tau} \tag{8}$$

The 'mean-excess-bandwidth' γ is defined as the mean value of the provisioning bandwidth minus the mean value of the real bandwidth:

$$\gamma = (1/t) \left(\sum_{\tau=1}^{t} \dot{T}(i,j)^{\tau} - \sum_{\tau=1}^{t} T(i,j)^{\tau} \right) \tag{9}$$

The 'mean-error' E is defined as the sum of the absolute values of the differences between the bandwidth estimates and the real bandwidth

$$E = \sum_{\tau=0}^{t} \left| \hat{T}(i,j)^{\tau} - T(i,j)^{\tau} \right| \tag{10}$$

The 'mean-squared-error' $\bar{E}^2$ is defined as

$$\bar{E}^2 = (1/t) \sum_{\tau=1}^{t} \left(\hat{T}(i,j)^{t} - T(i,j)^{t} \right)^2 \tag{11}$$

The results of this model, called the ARIMA(Q) model, are reported in Tables 1 and II. By changing the ARIMA-based filter coefficients and filter sizes, several variations of this model can be achieved.

B. *Traffic Estimation Exploiting Hourly History*

The traffic demands exhibit significant auto-correlations over hours, which the previous model does not exploit or capture. In this algorithm, two traffic estimates can be computed. To model a longer time horizon, the previous model can be extended to use multiple measurements of the change in traffic demands in the previous hour(s). The final traffic estimate can use a weighted average of both traffic estimates. The weights can change dynamically, based upon which estimate is more accurate.

A weighted-moving-average of the differences in traffic intensities over a window (the last hour) can be computed using a new filter z_h as follows:

$$\Delta_h = \sum_{\tau=0}^{3} z_h(\tau)(\bar{T}(i,j)^{t-\tau} - \bar{T}(i,j)^{t-\tau-1}) \tag{12}$$

for a weighting filter vector z_h. The second estimate of the traffic demand at time $t+1$ in the current day can be computed using:

$$\hat{T}(i,j)_h^{t+1} = T(i,j)^t + \Delta_h \tag{13}$$

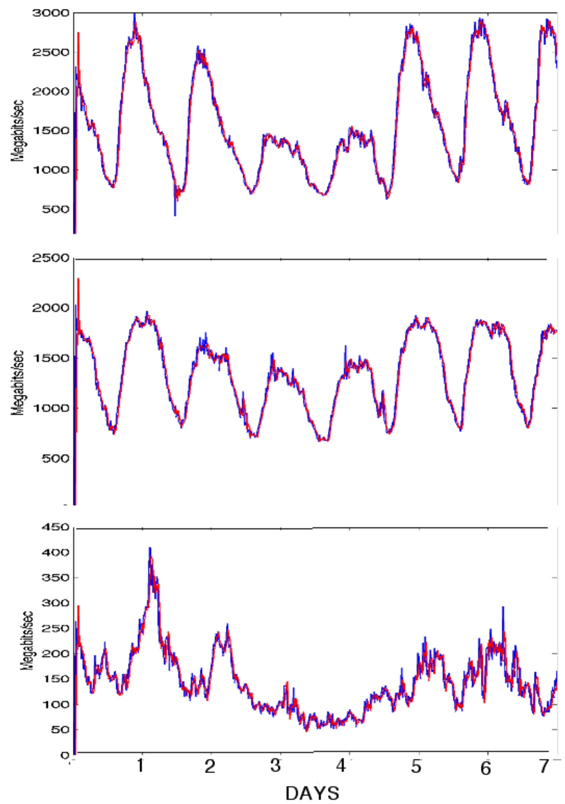

Fig. 4. Traffic estimates for Largest Traffic flows leaving a node, real (blue) and estimated (red).

The final traffic estimate can be computed using a weighted average of both estimates, using the existing weights:

$$\hat{T}(i,j)^{t+1} = w_q^t \cdot \hat{T}(i,j)_q^{t+1} + w_h^t \cdot \hat{T}(i,j)_h^{t+1} \tag{14}$$

To provision QoS bandwidth between a pair of cities, a standard deviation term can be added to the estimated traffic demand to yield the provisioning bandwidth:

$$\sigma_h^2 = Var(w(t-4t,1)) \tag{15}$$

$$\dot{T}(i,j)^{t+1} = \hat{T}(i,j)^{t+1} + w_q^t k_q \sigma_q + w_h^t k_h \sigma_h \tag{16}$$

where k_h and k_q are constants which determine the amount of excess bandwidth provisioned.

The real traffic demand will be measured after the time advances, and after estimate has been made. The weights can then be adjusted based upon the accuracy of the individual estimates: We compute the error terms as the absolute values of the differences between each estimate and the real traffic demand as follows:

$$\epsilon_q = \left| \hat{T}(i,j)_q^{t+1} - T(i,j)^{t+1} \right| \tag{17}$$

$$\epsilon_h = \left| \hat{T}(i,j)_h^{t+1} - T(i,j)^{t+1} \right| \tag{18}$$

The error terms can then used to update the weights to be used in the next measurement estimate:

$$w_q^{t+1} = \epsilon_h/(\epsilon_q + \epsilon_h) \tag{19}$$

$$w_h^{t+1} = \epsilon_q/(\epsilon_q + \epsilon_h) \quad (20)$$

The weights add more emphasis to the better estimate, and the sum of weights is unity. (Other weight functions can also be used.)

The results of this model, before weighting with the model ARIMA(Q), are reported in Tables 1 and II in the row labelled ARIMA(H). Results are shown for various values of the parameters k, which determine the amount of 'excess bandwidth' added into each provisioning-bandwidth estimate.

C. Traffic Estimation Exploiting Daily History

In this filter, 3 traffic estimates can be computed. The first 2 estimates can be computed using the equations presented above for the previous quarter-hour and previous hour. A third traffic estimate can also be computed, using a weighted-average of the differences in traffic intensities in the same hour interval in the previous day(s). A day represents 96 quarterly hour measurements. The final traffic estimate can be a weighted average of all three traffic estimates. The weights can change dynamically, based upon which estimate is more accurate.

A weighted-moving-average of the measured differences in the same hour interval of the previous day(s) can be computed using a new filter z_d as follows:

$$\Delta_d = \sum_{\tau=0}^{3} z_d(\tau)(\bar{T}(i,j)^{t-96-\tau+1} - \bar{T}(i,j)^{t-96-\tau}) \quad (21)$$

A third estimate of the traffic demand at time $t+1$ in the current day can be computed using:

$$\hat{T}(i,j)_d^{t+1} = T(i,j)^t + \Delta_d \quad (22)$$

(We can also use more quarter-hour measurements from each day in the model.) The final traffic estimate is computed using a weighted average of all three estimates, using the existing weights:

$$\hat{T}(i,j)^{t+1} = w_q^t \cdot \hat{T}(i,j)_q^{t+1} + w_h^t \cdot \hat{T}(i,j)_h^{t+1} + w_d^t \cdot \hat{T}(i,j)_d^{t+1} \quad (23)$$

A standard-deviation term based on the short-term variance for the same interval in the previous days can be computed:

$$\sigma_d^2 = Var(w(t-96-3, t-96, 1)) \quad (24)$$

The provisioning bandwidth is given by:

$$\dot{T}(i,j)^{t+1} = \hat{T}(i,j)^{t+1} + w_q^t k_q \sigma_q + w_h^t k_h \sigma_h + w_d^t k_d \sigma_d \quad (25)$$

where k_d, k_h and k_q are constants which determine the amount of excess bandwidth provisioned.

The real traffic demand will be measured after the time advances, and after estimate has been made. The weights can then be adjusted based upon the accuracy of the individual estimates: We compute the error terms between each estimate and the real traffic demand as follows, letting the symbol $\theta \in (q, d, h)$:

$$\epsilon_\theta = \left| \hat{T}(i,j)_\theta^{t+1} - T(i,j)^t \right| \quad (26)$$

The error terms can then used to update the weights to be used in the next measurement estimate:

$$w_\theta^{t+1} = (1/\epsilon_\theta)/(1/\epsilon_q + 1/\epsilon_h + 1/\epsilon_d) \quad (27)$$

(Other weight functions can also be used.) The weights add more emphasis to the better estimate, and the sum of weights is unity.

The results of this model, before weighting with the models ARIMA(Q) and ARIMA(H), are reported in Tables 1 and II in the row labelled ARIMA(D).

D. Traffic Estimation Exploiting Weekly History

The previous model can be extended to include more past history. Each week represents 96*7 = 672 measurements. A weighted-moving-average of the measured differences over the same hour interval in the last week can be computed using a new filter z_w as follows:

$$\Delta_w = \sum_{\tau=0}^{3} z_w(\tau) \left(\bar{T}(i,j)^{t-672-\tau+1} - \bar{T}(i,j)^{t-672-\tau}\right) \quad (28)$$

Another estimate of the traffic demand at time $t+1$ in the current day can be computed using:

$$\hat{T}(i,j)_w^{t+1} = T(i,j)^t + \Delta_w \quad (29)$$

The final traffic estimate can be computed using a weighted average of all 4 estimates, using the existing weights:

$$\hat{T}(i,j)^{t+1} = w_q^t \cdot \hat{T}(i,j)_q^{t+1} + w_h^t \cdot \hat{T}(i,j)_h^{t+1} + w_d^t \cdot \hat{T}(i,j)_d^{t+1} + w_w^t \cdot \hat{T}(i,j)_w^{t+1}$$

A standard-deviation term based on the short-term variance for the same interval in the previous week can be computed:

$$\sigma_w^2 = Var(w(t-672-3, t-672, 1)) \quad (30)$$

The provisioning bandwidth estimate is given by

$$\dot{T}(i,j)^{t+1} = \hat{T}(i,j)^{t+1} + w_q^t k_q \sigma_q + w_h^t k_h \sigma_h + w_d^t k_d \sigma_d + w_w^t k_w$$

where k_w, k_d, k_h and k_q are constants which determine the amount of excess bandwidth provisioned.

The real traffic demand will be measured after the time advances, and after estimate has been made. The weights can then be adjusted based upon the accuracy of the individual estimates: The error terms between each estimate and the real traffic demand are given as follows, letting the symbol $\theta \in (q, h, d, w)$:

$$\epsilon_\theta = \left| \hat{T}(i,j)_\theta^{t+1} - T(i,j)^{t+1} \right| \quad (31)$$

The error terms are then used to update the weights to be used in the next measurement estimate.

$$w_\theta^{t+1} = (1/\epsilon_\theta)/(1/\epsilon_q + 1/\epsilon_h + 1/\epsilon_d + 1/\epsilon_w) \quad (32)$$

The weights add more emphasis to the better estimate, and the sum of weights is unity.

The results of this model, before weighting with the models ARIMA(Q), ARIMA(H) and ARIMA(D) models, are reported in Tables 1 and II in the row labelled ARIMA(W).

TABLE I
TABLE 1. SATISFIED-BW FOR 5 TRAFFIC MODELS

Traffic Model	K=0	K=1	K=2
ARIMA(Q)	92.9%	97.1%	98.4%
ARIMA(H)	92.9%	97.1%	98.4%
ARIMA(D)	93.3%	97.2%	98.4%
ARIMA(W)	92.9%	97.1%	98.9%
ARIMA(Q,H,W,D)	94.0%	97.7%	98.9%

TABLE II
TABLE 2. EXCESS-BW FOR 5 TRAFFIC MODELS

Traffic Model	K=0	K=1	K=2
ARIMA(Q)	7.6%	20.0%	35.8%
ARIMA(H)	7.6%	20.0%	35.8%
ARIMA(D)	7.1%	20.1%	35.8%
ARIMA(W)	7.6%	20.0%	35.8%
ARIMA(Q,H,W,D)	6.1%	16.8%	30.0%

E. Results of Traffic Estimation

The results for the 4 traffic models are shown in Tables I and II. The results are based on the most intense traffic flows leaving each node, in a 3 week window. All 4 models, ARIMA(Q), ARIMA(H), ARIMA(D) and ARIMA(W), achieve very high 'mean-satisfied-bandwidths', between 93% and 99%. The row labelled ARIMA(Q,H,D,W) reports the results when all 4 estimates are weighted and combined, as described earlier. This model achieves the best performance.

Table II illustrates the 'mean-excess-bandwidth' for each model, for various values of K. There is no penalty associated with this figure, since in the proposed *Future Internet* network, bandwidth that is reserved but not used by the QoS class can be used by the Best-Effort traffic. The row labelled ARIMA(Q,H,D,W) reports the results when all 4 estimates are weighted and combined, as described earlier. This model achieves the best performance, minimizing the amount of excess-bandwidth.

V. CONCLUSIONS

Adaptive traffic estimation algorithms for an *Autonomic Future Internet* which can achieve can achieve *Essentially Perfect* link-utilizations and end-to-end QoS guarantees, along with significantly improved energy-efficiencies, have been proposed. The traffic estimation algorithms are based on several ARIMA-style filters operating in parallel, each estimating traffic using a certain time-scale, i.e., the quarter-hour, hour, day and week. An *Autonomic Controller* in each router can use the traffic estimates to provision bandwidth for QoS-enabled traffic flows between cities in the *Future Internet* backbone network, in anticipation of their arrival. An *Autonomic Controller* in each router can also use the traffic estimates to provision bandwidth for the *Differentiated-Services* traffic classes in each router in the existing *Best-Effort* Internet in anticipation of their arrival, thereby realizing an *Autonomic DiffServ Control Plane*. To test the algorithms, real traffic measurements taken every 15 minutes over a period of 4 months for a European backbone network are used. The traffic models are shown to be very accurate, provisioning sufficient bandwidth between pairs of cities with typically 93% ..99% success rates. The models can be extended in many ways. Statistically, the traffic demands on weekends (Saturday, Sunday) and holidays differ from weekdays, which the models could consider. Our models currently use moving-averages computed over a one-hour window in the previous day or previous week. The models can be extended to use moving-averages computed over several hours, over several days or several weeks.

REFERENCES

[1] L.G. Roberts, "A Radical New Router: The Internet is Broken - Lets Fix It", IEEE Spectrum, July 2009.
[2] S. Orlowski, R. Wessaly, M. Pioro, A. Tomaszewski, "SNDlib 1.0 - Survivable Network Design Library", Networks, Vol. 55, Issue 3, pp. 276-286, Oct. 2009.
[3] W.E. Leland, M.S. Taqqu, W. Willinger, D.V. Wilson, "On the Self-Similar Nature of Ethernet Traffic", IEEE/ACM Trans. Networking, Vol. 2, No. 1, Feb. 1994
[4] V. Anantharam, N. McKeown, A. Mekittikul and J. Walrand, "Achieving 100% Throughput in an Input Queued Switch", *Trans. Comm.*, vol. 47, no. 8, pp. 12601267, 1999.
[5] J.O. Kephart, D.M. Chess, "The Vision of Autonomic Computing", IEEE Computer, Jan. 2003
[6] A. Gunnar, M. Johansson, T. Telkamp, "Traffic Matrix Estimation on a Large IP Backbone - A Comparison on Real Data", ACM Int. Meas. Conf., Italy, Oct. 2004
[7] V.S. Frost, "Traffic Modeling for Telecommunications Networks", IEEE Comm. Mag. March 1994.
[8] C. Barakat, P. Thiran, G. Iannaccone, C. Diot, and P. Owezarski "Modeling Internet Backbone Traffic at the Flow Level", IEEE Trans. Signal Processing, Vol. 51, No. 8, Aug. 2003.
[9] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely, C. Diot, "Packet-Level Traffic Measurements from the Sprint IP Backbone", IEEE Network, 2003
[10] C. Fraleigh, F. Tobagi, and C. Diot, "Provisioning IP Backbone Networks to Support Latency Sensitive Traffic", IEEE INFOCOM 2003.
[11] K. Papagiannaki, N. Taft, Z.L. Zhang, and C. Diot, "Long-Term Forecasting of Internet Backbone Traffic", IEEE INFOCOM 2003.
[12] A. Soule, A. Lakhina ,N, Taft, K. Papagiannaki, K. Salamatian, A. Nucci, M. Crovella, C. Diot "Traffic Matrices: Balancing Measurements, Inference and Modeling", ACM Sigmetrics 2005
[13] N.K. Groschwitz and G.C. Polyzos, "A Time Series Model of Long-Term NSFNET Backbone Traffic", ICC 1994, New Orleans, May 1994
[14] C. Tran, Z. Dziong, "Kalman filter based capacity adaptation for network profit maximization", IEEE Int. Conf; Telecom. Network Strategy and Planning Symp. 2010
[15] J. Zhou, J. Yang, Y. Yang, H. Liu, M. Zeng, "Traffic matrix estimation based on a square root Kalman filtering algorithm", Int. Journal of Network Management archive, Vol. 18 Iss. 6, Nov. 2008
[16] T.H. Szymanski and D. Gilbert, "Provisioning Mission-Critical Telerobotic Control Systems over Internet Backbone Networks with Essentially-Perfect QoS", IEEE JSAC, Vol. 28, No. 5., June 2010
[17] T.H. Szymanski, "Video Multicasting in an Autonomic Future Internet with Essentially-Perfect Throughput and QoS Guarantees", 11th Int. Conf. on ITS Telecom. (NEW2AN 2011) , St. Petersburg, Russia, Aug. 2011, (and Springer Lecture Notes in Computer Science, Vol. 6869)
[18] I. Keslassy, M. Kodialam, T.V. Lakshman and D. Stilliadis, "On Guaranteed Smooth Scheduling for Input-Queued Switches", IEEE/ACM Trans. Networking, Vol. 13, No. 6, Dec. 2005