

Research Article

Automatic Bandwidth Adjustment for Content Distribution in MPLS Networks

D. Moltchanov

Institute of Communication Engineering, Tampere University of Technology, P.O. Box 553, 33101 Tampere, Finland

Correspondence should be addressed to D. Moltchanov, moltchan@cs.tut.fi

Received 2 May 2007; Revised 19 September 2007; Accepted 15 January 2008

Recommended by Zhu Li

Aggregates of real-time traffic may experience changes in their statistical characteristics often manifesting non stationary behavior. In multi protocol label switching (MPLS) networks this type of the traffic is assigned constant amount of resources. This may result in ineffective usage of resources when the load is below than expected or inappropriate performance when the load is higher. In this paper we propose new algorithm for dynamic resource adaptation to temporarily changing traffic conditions. Assuming that network nodes may reallocate resources on-demand using automatic bandwidth adjustment capability of MPLS framework, the proposed algorithm, implemented at ingress MPLS nodes, dynamically decides which amount of resources is currently sufficient to handle arriving traffic with given performance metrics. This decision is then communicated to interior MPLS nodes along the label switched path. As a basic tool of the algorithm we use change-point statistical test that signals time instants at which statistical characteristics of traffic aggregates change. The major advantage of the proposed approach is that it is fully autonomous, that is, network nodes do not need any support from hosts in terms of resource reservation requests. The proposed algorithm is well suited for traffic patterns experiencing high variability, especially, for non stationary type of the traffic.

Copyright © 2008 D. Moltchanov. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

Congestion is one of the major problems that degrades performance of networks. It occurs when network resources are insufficient for a current load or when load distribution is uncontrolled. The latter was the main problem of IP networks back in 1990s. Existing interior gateway protocols (IGPs) were topology-driven, took decision on routing packets based on network connectivity only and did not take into account resource requirements of traffic patterns. Traffic engineering (TE) capabilities of MPLS framework are intended to avoid the situation where congestion occurs as the result of inefficient resource allocation. TE attempts to provide best possible utilization of network resources avoiding noneven load distribution. Nowadays, MPLS is considered the vital part of QoS-aware networks.

Traffic patterns in packet-switched networks are characterized by deterministic trends similar to those found in classic telephone traffic. Particularly, it was demonstrated that there are clear daily variations in statistical parameters

of traffic aggregates and these variations can be specific for different services [1]. They are often caused by preference of users to use a certain network service in a specific time of the day. Authors in [1] also noticed that there is a clear indication of the busy hour in link usage patterns. Irregular changes in statistical parameters of traffic aggregates were also reported in [2–4]. In [5], authors highlighted that operators' pricing scheme may also produce changes in traffic patterns resulting in additional source of nonstationarity. Besides commonly believed self-similar and long-range dependent properties (see [6–8] among others), nonstationarity is one of the major reasons for a traffic pattern to exhibit high variability.

In modern QoS-aware networks, resources for traffic aggregates are often allocated statically based on busy hour traffic expectations. In this case, high variability of a traffic pattern requires significant overprovisioning of network resources. Practically, high variability implies that there are large bursts in a traffic pattern. To serve such traffic with given performance metrics, sufficient amount of resources must be provided during this period. However, there are also

long time spans during which the local average of the traffic pattern stays well below the global average. This implies that for a traffic pattern exhibiting high variability static resource allocation leads to ineffective usage of network resources. We also note that the busy hour traffic volume may not be known in advance making the decision about the required resources completely unclear.

The network always has up-to-date view of traffic aggregates. The straightforward way to deal with overprovisioning or underprovisioning of resources is to allow network nodes to dynamically choose the amount of resources that is sufficient for current arrival statistics of a traffic aggregate. The rest of resources can be temporarily assigned to other traffic aggregates. To deal with this problem, MPLS networks provide automatic bandwidth adjustment capability. It allows an MPLS label switched path (LSP) to automatically adjust bandwidth allocation based on its traffic load measured over the sampling interval of a certain length. Unfortunately, automatic bandwidth adjustment procedure does not take into account that statistical characteristics of traffic patterns may vary even during a single sampling interval. The length of the sampling interval is another important issue that affects performance of the resource allocation and requires additional administrative efforts. Dealing with time-varying traffic patterns, fully automatic bandwidth adjustment scheme that requires no additional administrative effort would be of a great benefit for network operators.

In this paper, we propose resource description and allocation scheme that dynamically reallocate resources to traffic aggregates in MPLS networks. Assuming that network nodes may reallocate resources on-demand using automatic bandwidth adjustment capability of MPLS framework, the proposed algorithm decides which amount of resources is currently sufficient to handle arriving traffic with given performance guarantees. We describe the structure of the proposed system and demonstrate its applicability using real traffic traces. As a basic tool of the algorithm, we use change-point statistical test. It allows to automatically determine whether statistical characteristics of a traffic pattern change and, if so, estimate new parameters of the traffic pattern. The major advantage of the proposed approach is that it is fully autonomous, that is, the network does not need any support in terms of explicit resource reservation requests. The proposed algorithm is especially well-suited for nonstationary type of the traffic whose statistical parameters change in time. Presented numerical results reveal that the proposed approach may provide significant resource savings.

The rest of the paper is organized as follows. In Section 2, we briefly review automatic bandwidth allocation capability of MPLS networks. In Section 3, we consider dynamic nature of aggregated video traffic. Model for covariance-stationary behavior of aggregated video traffic is also proposed there. Change-point statistical test for detecting changes in arrival statistics is introduced in Section 4. The dynamic resource allocation system is proposed in Section 5. Numerical examples are given in Section 6. Conclusions are drawn in the last section.

2. MPLS NETWORKS

2.1. Traffic engineering in MPLS

Modern QoS-aware networks such as DiffServ, MPLS, or joint DiffServ/MPLS are specifically designed to be flexible enough to reallocate network resources in the best possible way, such that the required performance is provided using minimum amount of resources. Although MPLS is not considered a QoS framework for IP networks, it provides a number of advantageous features to network operators. According to MPLS, data are transmitted along the so-called label switched paths (LSPs) that can be explicitly chosen by network operators [9]. For this purpose, MPLS assigns short labels to network packets that describe how to forward them through the network. This feature enables effective traffic engineering (TE) capabilities [10] compared to classic approach that adopts usage of interior gateway protocol (IGP) metrics. Label switched paths (LSPs) are computed in traffic engineering databases (TED, [10]) in ingress routers that are filled by IGP advertisements. For this reason, IGP protocols were extended to support advanced metrics such as available bandwidth on a link [11, 12]. LSPs are then established using either resource reservation protocol modified to support traffic engineering (RSVP-TE, [13]) or label distribution protocol with constrained routing (CR-LDP, [14]).

Traffic engineering capabilities of MPLS allow to control the path that data packets follow in the network avoiding standard routing strategy of IGP protocols. They are intended to deal with the situation where congestion occurs as the result of inefficient resource allocation. TE tries to provide best possible utilization of network resources avoiding noneven load distribution. This is achieved by real-time monitoring of the traffic load on each network element and further tuning of traffic management attributes, routing parameters, and resources constraints.

2.2. Automatic bandwidth adjustment

Traffic aggregates in DiffServ, MPLS, and joint MPLS/DiffServ networks are still described statically using the token bucket mechanism. Automatic bandwidth adjustment capability of MPLS networks allows an LSP to automatically adjust its bandwidth allocation based on the measured traffic load. It works as follows. During the sampling interval of a certain length, the average bandwidth is monitored. At the end of the interval, an attempt is made to signal a new LSP with the bandwidth allocation set to the average value for the preceding sampling interval. If a new path is established, the original path is removed and the LSP is switched to the new path. If a new path is not established, the LSP continues to use its current path until the end of the next sampling interval, when another attempt is made.

While the approach described above is considered to be automatic in nature, it does not take into account possible time-varying behavior of traffic patterns. When the sampling interval is too large, there can be multiple changes in the mean value of traffic observations that may severely

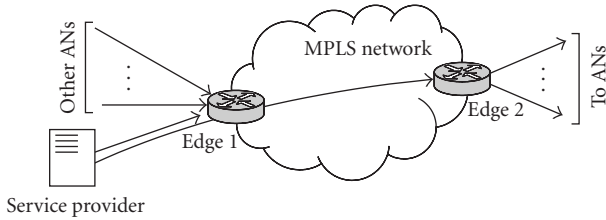


FIGURE 1: Reference configuration of the service.

bias the statistics. On the other hand, too short sampling interval may burden the network with excessive signaling information. The approach that triggers LSP reestablishment when statistical characteristics of a traffic patterns do change could provide trade-off between the amount of signaling information and accuracy of bandwidth allocation.

2.3. Service configuration

Reference configuration of the service we consider is presented in Figure 1, where AN stands for access network. We assume that there is a network operator providing network resources for a service provider. Service provider provides a QoS-constrained video distribution service to a number of users across the network. We assume that there is a specifically established LSP for video traffic aggregate and routers along the path of the traffic aggregate implement MPLS framework with dynamic bandwidth adjustment capability. The network operator is required to provide guarantees to this traffic aggregate in terms of mean losses and delays.

3. TIME-VARYING NATURE OF AGGREGATED VIDEO TRAFFIC

To represent traffic arrival process, we use video traffic traces available from the University of Berlin [15]. We consider video traffic using the granularity of a second. Although we use H.263 VBR traces, we have checked that conclusions stated in this paper remain valid for MPEG-4 VBR traces from the same traffic archive and MPEG-1 VBR sequences from University of Wurzburg [16].

3.1. Traffic aggregate: varying number of sources

We generated traffic traces simulating behavior of the video distribution system as explained below. We assume that at a certain instant of time t_0 there are no active sessions and this is the time instant when the system enters its operational state. After t_0 , session requests start to arrive to the system. Each arrival requests an arbitrary video sequence. We used uniform distribution over all available traces to determine the requested sequence. Interarrival times of sessions are geometrically distributed with a certain probability P . We assumed that session arrival process is time-varying meaning that P varies in time. This is a natural assumption for any service where the session arrival rate is different for different time of the day. We simulated 50 session arrivals. First 15

and last 15 arrivals were drawn from geometrical distribution with $P = .001$, 20 arrivals in the middle occur according to geometrical distribution with $P = .005$. Note that these parameters allow to mimic behavior of the system prior, during and after the busy hour.

Time-series of generated traces are shown in Figure 2. Observing these traces, it is natural to expect that statistics of the aggregated traffic are time-varying. Indeed, when the rate of session arrivals varies in time, it is unrealistic to assume that aggregated traffic may converge to covariance stationary process except for some segments during which the number of active sessions remains constant. For other choices of P and different number of generated sessions, traces look qualitatively similar meaning that their statistical parameters do vary in time. Note that statistical characteristics of traffic aggregates may also vary due to other phenomena including long-range dependent, self-similar or nonstationary properties of individual streams. In this paper, the actual cause of variability is of no interest as long as change detection mechanism is able to detect points at which changes occur and subsequently estimate current traffic statistics.

An important property of aggregated traffic is that changes in the mean value lead to changes in the variance and vice versa. Figure 3 highlights this behavior. As one can observe, this dependency is almost linear allowing to track changes in one parameter only, either mean or variance.

3.2. Traffic aggregate: fixed number of sources

Consider statistical characteristics of the aggregated traffic when the number of multiplexed sources is kept constant. To obtain aggregated traffic patterns, we arbitrarily choose 9 traces out of the archive and multiplex them synchronizing starting times of all traces. We limited all traces to 1200 seconds.

Time-series of six aggregated traces are shown in Figure 4. Observing these traces, one may suggest that statistics of the aggregated traffic may not significantly vary in time. Indeed, when the number of video traffic sources is fixed and relatively large, it is straightforward to assume that the aggregated traffic may converge to covariance stationary process with normal distribution [5].

To support our hypothesis about covariance stationarity, we carried out a number of tests. First, we divided each trace into 6 nonoverlapping segments each of which contains exactly 200 observations. Mean values of these segments are shown in Figure 5. One may see that the mean value of single traces does not change in time significantly. For all traces, the ranges of mean values are less than 20% of global means of respective traces. These variations can be attributed to positive autocorrelation found in these traces. Finally, we tested our traces for homogeneity. To do so, each trace was divided into two samples having the same number of observations. In our case, this corresponds to 600 observations. Then, we applied χ^2 test for homogeneity of two samples. This test allows to check hypothesis H_0 that two parts have the same distribution against alternative hypothesis, H_1 , that they have different distributions. Performed tests revealed that with

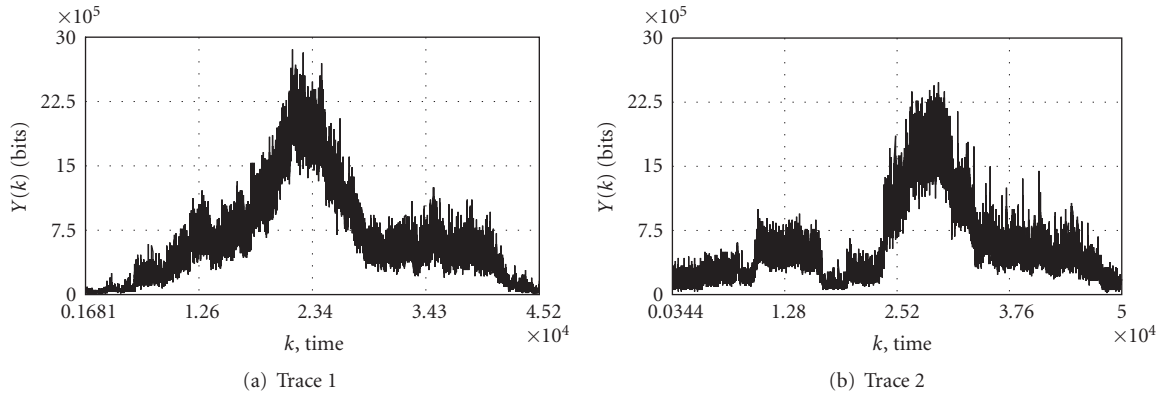


FIGURE 2: Aggregated traffic from varying number of sources.

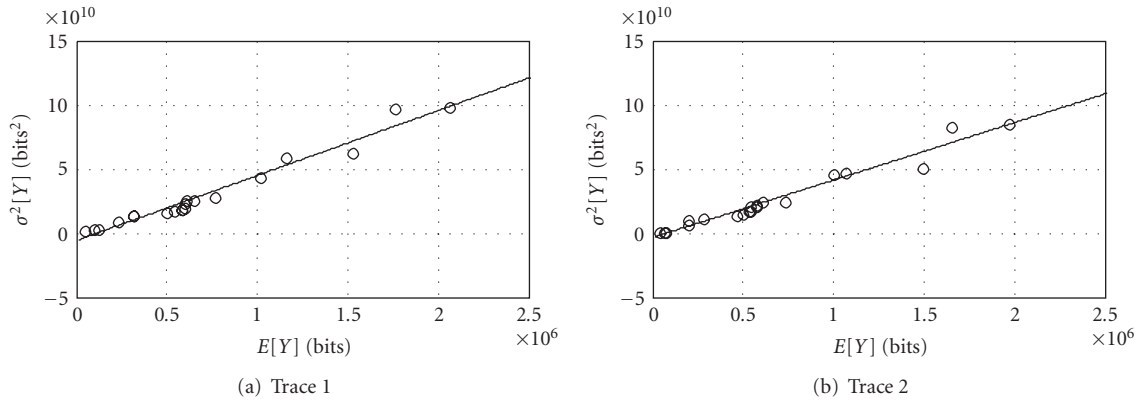


FIGURE 3: Dependence between mean and variance of aggregated traffic.

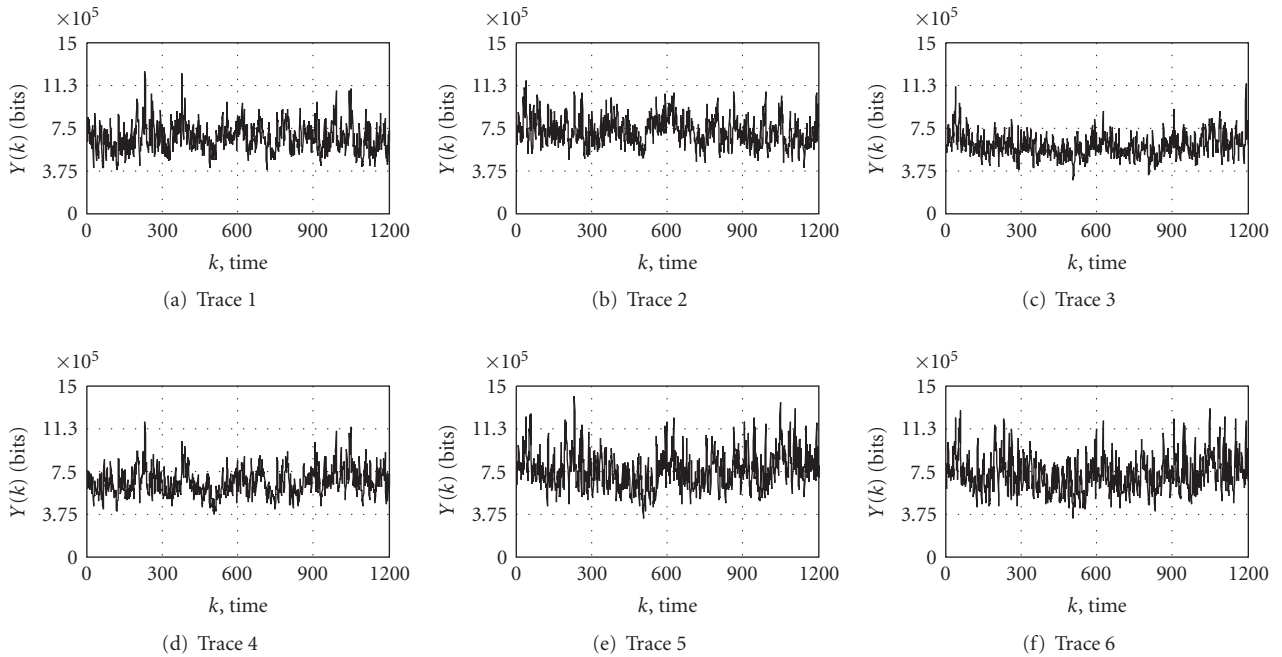


FIGURE 4: Aggregated traffic traces from a fixed number of sources.

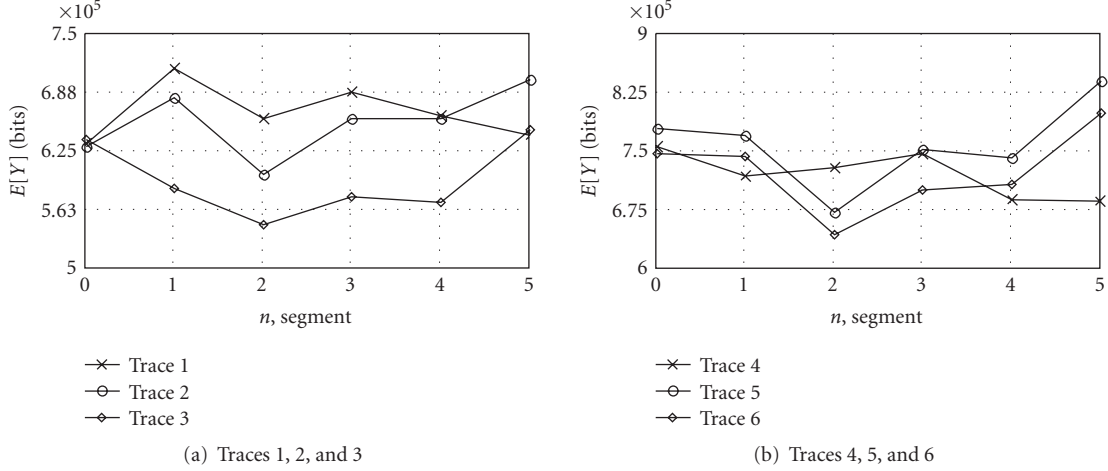


FIGURE 5: Mean values of segments.

level of significance $\alpha = 0.1$ two parts of the same trace have the same distribution.

Histograms of relative frequencies of considered traces and their approximations by normal distributions are shown in Figure 6, where $f_{i,Y}(\Delta)$, $i = 1, 2, \dots$ is the relative frequency corresponding to i bin, $\Delta = (\max_{v_i} Y(i) - \min_{v_i} Y(i))/m$ is the width of the histogram bin. These approximations suggest that aggregated traffic from fixed number of sources is normally distributed. The χ^2 goodness-of-fit test performed with the level of significance $\alpha = 0.1$ confirmed this hypothesis.

Empirical normalized autocorrelation functions (NACFs) for all traces are shown in Figure 7. One may note that the memory of the process is short and limited to few lags. We approximate this behavior using a single geometrical term, for example, $y(i) = K_Y(1)^i$, $i = 0, 1, \dots$. This approximation exactly captures lag-1 values of NACFs of empirical processes and does not significantly overestimate or underestimate autocorrelation coefficients for larger lags.

Presented results allow us to assume that when the number of multiplexed sources remains constant the aggregated video traffic composes realization of covariance stationary stochastic process. We note that our tests do not allow us to be statistically strict regarding covariance stationarity. Unfortunately, there are no universal and effective methods to statistically test whether given observations are stationary or not. However, Carried out tests confirm that the marginal distribution is normal and depends on the number of multiplexed traces only NACF tends to zero as time progresses confirming that the process is ergodic. All these conditions are necessary for an underlying process to be covariance stationary. In Section 6, we carry out another test that also confirms our hypothesis.

3.3. Model for aggregated traffic

To model aggregated traffic from fixed number of sources, we use covariance stationary autoregressive process of order

one, AR(1). This process has normal marginal distribution and geometrically decaying ACF allowing us to suggest that it may produce fair approximation of empirical data [17]. These two properties of traffic patterns were found to produce the major impact on their service performance in a network (see [18, 19] among others).

A process is said to be autoregressive of order one if it is given by

$$X(n) = \phi_0 + \phi_1 X(n-1) + \epsilon(n), \quad n = 0, 1, \dots, \quad (1)$$

where ϕ_0 and ϕ_1 are some constants, $\{\epsilon(n), n = 0, 1, \dots\}$ are independently and identically distributed random variables having the same normal distribution with zero mean and variance $\sigma^2[\epsilon]$.

For (1) to be weakly stationary, it is sufficient to have $\phi_1 \neq 1$. In this case,

$$E[X(n)] = \mu_X, \quad \text{Cov}(X_0(n), X_i(n+i)) = \gamma_X(i), \quad (2)$$

where μ_X , $\gamma_X(i)$, $i = 0, 1, \dots$ are some constants.

Mean, variance and covariance of AR(1) are related to ϕ_0 , ϕ_1 , and $\sigma^2[\epsilon]$ as

$$\mu_X = \frac{\phi_0}{1 - \phi_1}, \quad \sigma^2[X] = \frac{\sigma^2[\epsilon]}{1 - \phi_1^2}, \quad \gamma_X(i) = \phi_1^i \gamma_X(0). \quad (3)$$

Parameters of AR(1) models are related to statistical data as

$$\phi_1 = K_Y(1), \quad \phi_0 = \mu_Y(1 - \phi_1), \quad \sigma^2[\epsilon] = \sigma^2[Y](1 - \phi_1^2), \quad (4)$$

where $K_Y(1)$, μ_Y , and $\sigma^2[Y]$ are estimates of lag-1 autocorrelation coefficient, mean, and variance, respectively.

4. CHANGE-POINT STATISTICAL TESTS

4.1. Basic principles

To differentiate between fluctuations of the aggregated traffic around a constant mean and those variations caused by

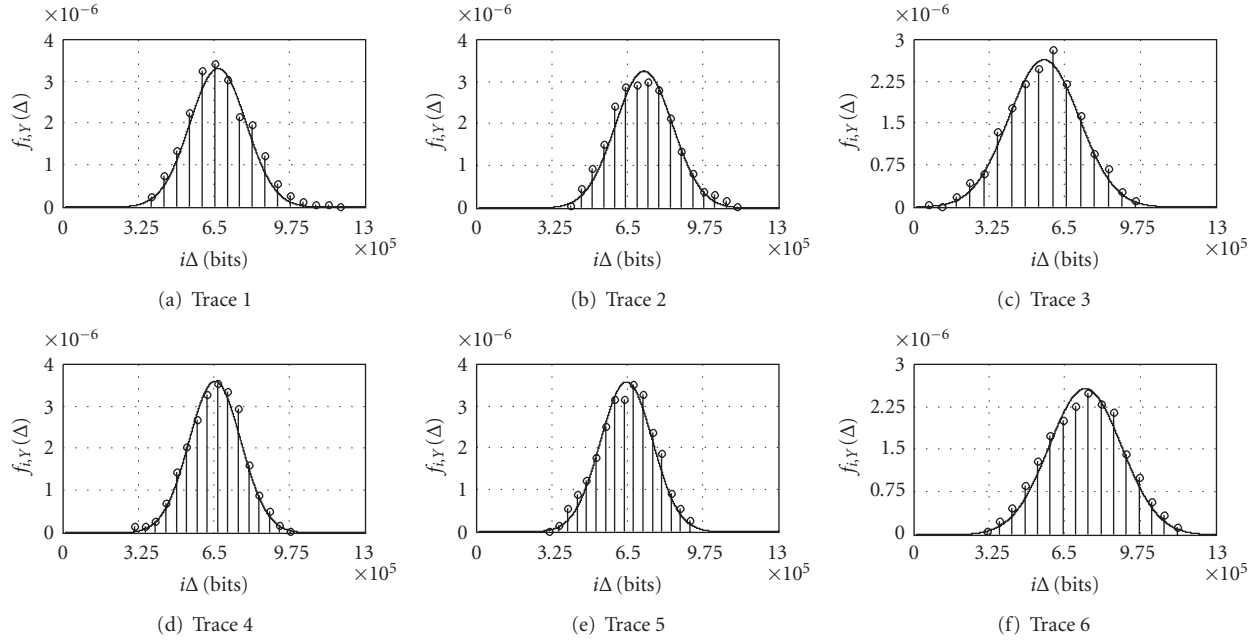


FIGURE 6: Histograms and their approximations by normal distribution.

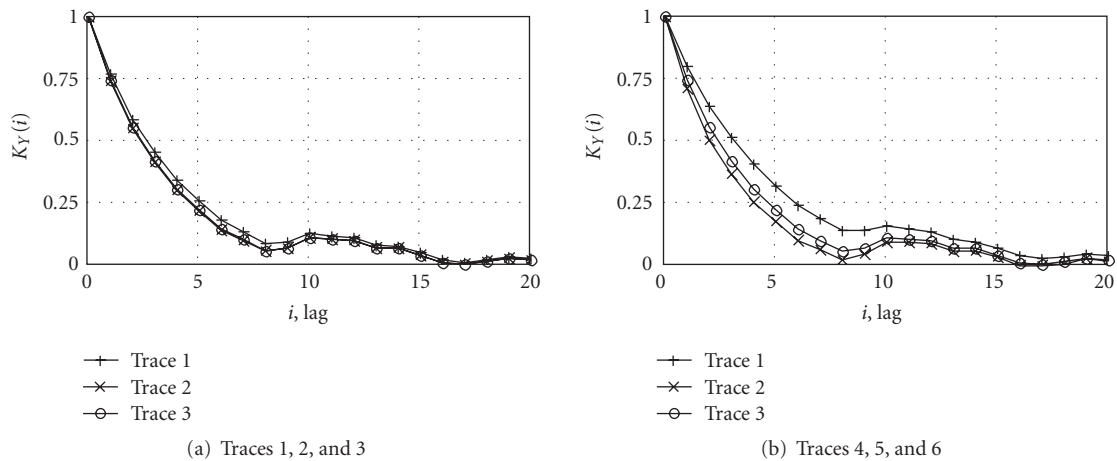


FIGURE 7: Normalized autocorrelation functions of traces.

changes in the mean value, we propose to use change-point statistical tests. There are a number of change-point detection algorithms developed to date. The common approach to deal with this task is to use control charts including Shewhart, cumulative sum (CUSUM), or exponentially-weighted moving average (EWMA) charts. The idea of control charts is to classify all causes of deviation from a target value into two groups. These are common causes and special causes. Deviation due to common causes is the effect of inherent causes affecting a given process. Special causes are not the part of the process, occur accidentally, and affect the process significantly. Control charts signal the point at which special causes occur using two control limits. If values of a certain function of initial observations are between them, process is in control. If some value falls outside, the process is out of control.

For detecting changes in aggregated traffic, we assume the following. We consider that common causes of deviation are those resulting in inherent stochastic nature of the multiplexed traffic from fixed number of video sources. Special causes are all those causing changes in parameters of in control processes. For example, among those are arrivals of new sessions adding new traffic and completions of ongoing ones subtracting some traffic. The control procedure is as follows. Initially, a control chart is parameterized using estimates of parameters of the aggregated traffic process. When change in a parameter occurs, a new process is considered as in control and the control chart is reparameterized according to statistics of this process.

Change-point tests often require observations to be independent. We have seen that aggregated video traffic is characterized by positive correlation. Autocorrelation makes

control charts less sensitive to changes in the mean value. For detecting changes in the mean value of autocorrelated processes, two approaches have been proposed. According to the first approach, control limits of charts are modified to take into account autocorrelation properties. The idea of the second approach is to fit observations using a certain time-series model and subsequently test residuals. If the model fits empirical data well, residuals are uncorrelated and control charts for independent observations can be used. Performance of change-point statistical tests for autocorrelated data has been compared in [20, 21]. It was shown that modified control charts on initial observations perform better when autocorrelation is positive. Due to this reason, we use the first approach.

4.2. EWMA control charts

Let $\{Y(n), n = 0, 1, \dots\}$ be a sequence of initial observations. The value of EWMA statistic at the time n , denoted by $L_Y(n)$, is given by

$$L_Y(n) = \gamma Y(n) + (1 - \gamma)L_Y(n - 1), \quad (5)$$

where parameter $\gamma \in (0, 1)$ is constant.

The reason to use EWMA statistical test is that it takes central part among other control charts. Although, according to (5), the most recent value always receives more weight in computation of $L_Y(n)$, the choice of γ determines the effect of previous observations of the process on the current value of EWMA statistics. Indeed, when $\gamma \rightarrow 1$ all weight is placed on the current observation, $L_Y(n) \rightarrow Y(n)$, and EWMA statistics degenerate to initial observations. As a result, EWMA control chart behaves like Shewhart one [22]. On the other hand, when $\gamma \rightarrow 0$ the current observation gets only a little weight, but most weight is assigned to previous observations. In this case, EWMA control chart behaves similar to CUSUM one [23]. Summarizing, EWMA charts provide more flexibility at the expense of additional complexity in determining one more parameter γ . Due to inherent flexibility, we use EWMA control charts.

Assume that initial observations $\{Y(n), n = 0, 1, \dots, N\}$ are taken from covariance stationary process with mean $E[Y]$ and variance $\sigma^2[Y]$ and can be well represented by AR(1) process. If $L_Y(0) = E[Y]$, it is easy to see that $E[L_Y] = E[Y] = \mu_Y$ when $n \rightarrow \infty$. The approximation for variance of $\{L_Y(n), n = 0, 1, \dots\}$ is given by [20]

$$\sigma^2[L_Y] = \sigma^2[Y] \left(\frac{\gamma}{2 - \gamma} \right) \left(\frac{1 + \phi_1(1 - \gamma)}{1 - \phi_1(1 - \gamma)} \right), \quad (6)$$

where ϕ_1 is the parameter of AR(1) process.

The control limits $E[L_Y] \pm C_Y(n)$ are given by

$$E[L_Y] \pm C_Y(n) = E[L_Y] \pm k\sigma[Y] \sqrt{\left(\frac{\gamma}{2 - \gamma} \right) \left(\frac{1 + \phi_1(1 - \gamma)}{1 - \phi_1(1 - \gamma)} \right)}, \quad (7)$$

where k is a design parameter whose values are tabulated.

4.3. Parametrization of EWMA charts

To parameterize the EWMA control chart, a number of parameters have to be provided. Firstly, parameter γ determining the decline of weights of past observations should be set. The values of k and γ determine the wideness of control belts for a given process with certain $\sigma^2[Y]$ and μ_Y . These four parameters affect behavior of the so-called average run length (ARL) value that is used to determine efficiency of a certain change detection procedure. ARL is defined as the average number of observation up to the first out of control signal. There are a number of methods to compute ARL value for given γ and k . Tabulated values of in control ARL can be found in [20, 21]. Finally, μ_Y and $\sigma^2[Y]$ are not usually known in practice and must be estimated from empirical data. Therefore, estimates of μ_Y and $\sigma^2[Y]$ should be used in (7). As a result, for real-time implementation of EWMA test, there should always be a certain warmup period involving estimation of the mean. Finally, the first value of EWMA statistics is usually set to the global mean of observations or, if unknown, to the estimate of mean.

5. RESOURCE ALLOCATION SYSTEM

5.1. Functionality of the system

The proposed resource description and allocation system should be implemented in ingress MPLS routers. We assume that ingress routers conform to MPLS specifications. The only important difference compared to MPLS functionality is the resource controller. The purpose of this controller is to monitor arriving traffic of different LSPs for possible changes in its statistical characteristics and manage the resource allocation for a given behavior aggregate. Another responsibility of the controller is to estimate the amount of resources required to serve incoming traffic with given performance metrics. It is important that this controller does not change the traffic pattern allowing it to proceed unaltered to the output port. We also note that no changes to interior nodes are required and they must strictly follow MPLS specifications.

The structure ingress node is shown in Figure 8. It is intended to operate as follows. Traffic policing unit is statically parameterized such that the highest possible load a content provider intends to pay for is allowed to enter the network. Those packets that conform to this specification proceed to the resource controller. This element monitors conforming traffic for possible changes in its statistical characteristics and estimates the amount of resources required to serve it with given performance metrics. Performance metrics must be configured manually. When change in traffic statistics occurs, resource controller estimates new resource allocation and advertises it to the resource allocation mechanism at the output port and to the MPLS TED associated with ingress node. RSVP-TE is then used to update resource allocation at all nodes along the path of LSP.

The structure of the resource controller is shown in Figure 9. Two major components of this system are the

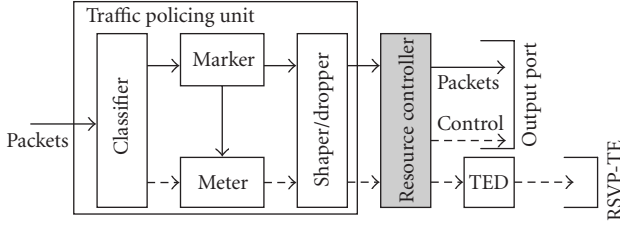


FIGURE 8: The structure of the ingress MPLS node.

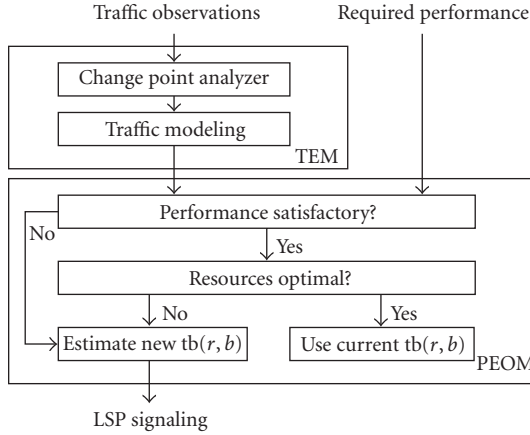


FIGURE 9: The structure of EWMA resource controller.

real-time traffic estimation module (TEM) and the performance evaluation and optimization module (PEOM). The system operates as follows. The TEM is responsible for detecting changes in arriving traffic statistics and dynamic estimation of the traffic state in terms of AR(1) model. To achieve that, traffic statistics must be observed in real-time and fed to the input of the change-point analyzer. Change-point analyzer tests incoming observations for changes in the mean value using EWMA statistical test. It is important to note that the change-point analyzer must signal the point when a change in arriving traffic statistics is detected. Otherwise, no actions must be taken by the TEM except for traffic monitoring. If a change is detected, traffic statistics are computed during the warmup period. Then a new model of the traffic is parameterized in the modeling block and fed to the input of PEOM.

According to the design of PEOM, traffic model is fed to the input of the decision module. Taking the required performance at the IP layer as another input, this module decides whether the current performance is satisfactory. If the performance is satisfactory, the system must further decide whether the amount of resources for a current state of the traffic is optimal. In order to take these two decisions, the module containing the performance evaluation framework is activated and supplied with the current traffic model. This module may implement the performance evaluation framework or just contain a set of precomputed performance curves corresponding to a wide range of traffic statistics. Due to the real-time nature of the performance control system, the latter approach is preferable.

Theoretically, when the input traffic statistics change, performance provided to aggregated traffic or optimal resource allocation providing a certain performance level or both change. However, in practical applications we always have a finite granularity of performance parameters and, therefore, we have to check whether a new traffic model does not satisfy performance requirements of aggregated traffic or results in not optimal resource allocations. If the performance is predicted to be unsatisfactory, the current traffic model is used to decide which resource allocation parameters provide both the required performance level and optimal usage of resources. Finally, if the resource allocation is not optimal but the performance is satisfactory, new optimal resource allocation must be computed. Obtained resource allocation parameters are then used till the next change in traffic statistics.

5.2. Estimating resource allocation

As a descriptor of the amount of resources required by an LSP, token bucket is used. The actual amount of resources in terms of the buffer space and outgoing link rate share is inferred from token bucket parameters at each node. According to the token bucket mechanism, the amount of traffic allowed in the time interval $[0, t]$ is upper bounded by $A(t) = rt + b$, $t \geq 0$, where token rate r is related to the outgoing link rate share, and bucket size b is related to the buffer space. To parameterize a token bucket, we have to find a Fair (r, b) satisfying performance criteria. Usually, there are infinite number of pairs (r, b) satisfying the required loss performance. However, there is always upper bound on b that also satisfies delay requirements. There are a number of approaches to estimate token bucket parameters using statistics of arrival process. Approaches range from approximations providing token bucket parameters for worst case traffic scenarios to exact ones involving solution of equivalent queueing systems [24, 25]. We are interested in simple approach providing a feasible option for online implementation.

To estimate parameters of the token bucket, we use overflow theory. Let $\gamma(t)$ be the cumulative arrival process and let $A(s, t)$ be the amount of work arriving in the interval $(s, t]$. The probability that the queue length exceeds b is then

$$Pr\{Q > b\} = Pr\left\{\sup_{t \geq 0} (\gamma(t) - rt) > b\right\}, \quad (8)$$

where r is the link rate share assigned to a traffic aggregate.

Using lower bound approximation, we get

$$Pr\left\{\sup_{t \geq 0} (\gamma(t) - Ct) > b\right\} \geq \sup_{t \geq 0} Pr\{\gamma(t) > b + Ct\}. \quad (9)$$

Denoting $F_t(x) = Pr\{\gamma(t) \leq x\}$, we get

$$Pr\{Q > b\} \geq \sup_{t \geq 0} Pr\{\gamma(t) > b + Ct\} = \sup_{t \geq 0} (1 - F_t(b + Ct)). \quad (10)$$

Note that the bucket size, b , should be set such that the delay is equal to or less than the maximum allowable

delay in a network element. Thus the task in (10) reduces to finding a suitable link rate share that should be assigned to a traffic aggregate. When $F_r(x)$ is normal, the following approximation can be used [26]:

$$r = E[X] + \alpha\sigma[X], \quad \alpha = \sqrt{-2 \ln \epsilon - \ln 2\pi}, \quad (11)$$

where ϵ is the buffer overflow probability.

5.3. The immediate update mechanism

According to our algorithm, new resource reservation should only be advertised to network elements when the resource controller already estimated a new allocation. This information is available at the end of the warmup period. For some traffic patterns (e.g., VoIP aggregates), this period can be long enough to receive inadequate treatment in a network. To ensure that the proposed algorithm continuously provides the desired level of performance, we propose to use following adaptive mechanism. When increase in the mean value is detected, a new resource allocation (r_+, b_+) is immediately estimated and advertised to the output buffer and network elements along the path of the LSP. Since at the beginning of the warmup period we do not know traffic statistics yet, the choice of (r_+, b_+) is somewhat arbitrary. One possible way is to estimate new resource allocation using mean $(E[Y] + C_Y)$ and variance, $\sigma^2[Y]$, where $E[Y]$, C_Y , and $\sigma^2[Y]$ are the mean, control limit, and variance of previous in control segment, respectively. When the warmup period expires and statistics of new in control process are available, new resource allocation is computed and advertised.

The same algorithm can be used when the mean value decreases. Another approach is to advertise new resource allocation at the end of the warmup period only. Indeed, since the mean value decreases we still ensure that appropriate performance is continuously provided. Note that the immediate update mechanism results in additional signaling load consisting of RSVP-TE reservation updates and subsequent IGP-TE resource allocation advertisements [27].

6. NUMERICAL EXAMPLES

6.1. Aggregated traffic: fixed number of sources

In Section 2, we assumed that aggregated traffic from fixed number of video traffic sources composes realization of the covariance stationary process. Let us now provide one more reason for this conclusion applying EWMA change-point test to first two traces demonstrated in Figure 4. We already found that this traffic is normally distributed and there is significant lag-1 autocorrelation. Due to these properties, control limits are computed according to (7). The warmup period used to compute control limits was set to 50 observations. EWMA statistics for different values of γ and k are shown in Figure 10. Note that $k = 3$, $\gamma = 0.01$ correspond to 137.91 in control ARL for the first trace and 175.24 in control ARL for the second trace. Parameters $k = 3$ and $\gamma = 0.001$ correspond to 956.68 in control ARL for

the first trace and 1286.97 in control ARL for the second one. One can observe that no change in the mean value of traffic observations is detected even though the ARL values for $k = 3$ and $\gamma = 0.01$ are relatively small. These results suggest that the mean value of the aggregated traffic from fixed number of sources does not vary in time as required by covariance stationarity. Similar results have been obtained for other traces.

Note that for both traces and all values of γ , the parameter k was set to 3. In general, to match a given ARL value, different values of γ require different values of k . It was also found that usage of tabulated values may lead to many out of control signals even when there are no new session arrivals or departures. One of the reasons is that the monitored process may not be exactly covariance stationary and may occasionally contain some extreme observations (outliers). These observations may be of local significance only and may not affect the future evolution of the monitored process as well as the service process of arriving traffic. Since the EWMA change detection is assumed to operate in real time, each out of control signal starts a new warmup period. During this period, new control limits are estimated and the process cannot be monitored. From this point of view, we should detect only those changes that occur for sure.

Results presented in this subsection provide additional necessary condition for covariance stationary of the multiplexed traffic from fixed number of video sources. Our observations stay in control in EWMA chart constructed for AR(1) process suggesting that they can be realization of this process. Thus EWMA chart can be seen as a tool for testing stationarity of observations.

6.2. Aggregated traffic: varying number of sources

Consider now aggregated traffic from varying number of sources shown in Figure 2. EWMA statistics computed for these traces are shown in Figures 11(a) and 11(b) ($k = 3$, $\gamma = 0.001$). Figures 11(c) and 11(d) demonstrate the same statistics for first several session requests, where solid horizontal lines represent control limits and boxes represent session arrivals. The first box indicates the time instant when second session request arrives to the system.

Consider EWMA statistics in detail. We started the control chart when the first session arrives. It occurred at 1681 second for trace 1 and at 334 second for trace 2. The warmup period used to compute statistics of observations and control limits of charts was set to 50 observations. One may observe from Figures 11(c) and 11(d) that both processes stay in control when the second sessions arrive and remain in control until new traffic adds up to EWMA statistics and changes are detected. Note that there are gaps before EWMA statistics exceed the upper control limit for both traces. This is due to the memory of EWMA statistics that allows to avoid false signals but worsens reactive properties of the chart. In general, when the value of γ gets smaller, this interval becomes larger while the probability of false change detection decreases.

EWMA statistics computed for our traces with $k = 3$ and $\gamma = 0.001$ are shown in Figures 12(a) and 12(b). The same

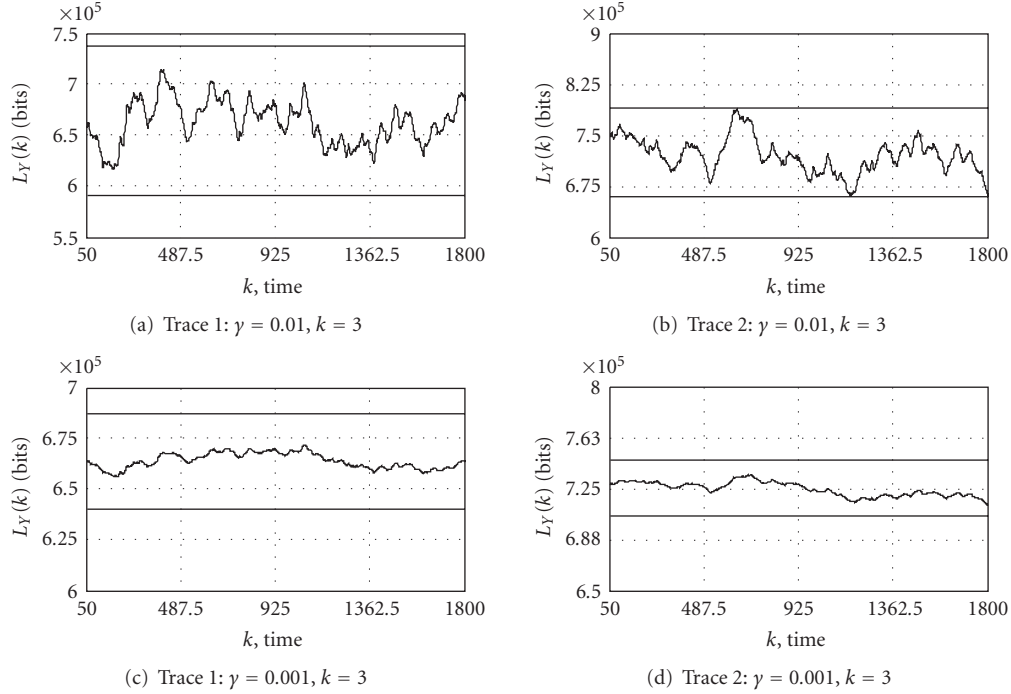


FIGURE 10: EWMA statistics for aggregated traffic: fixed number of sources.

statistics for first several session arrivals are shown in Figures 12(c) and 12(d). Note that changes in mean values of the aggregated traffic were successfully detected for these values of γ and k .

6.3. Dynamic resource allocation

The proposed dynamic resource allocation system tries to provide bandwidth savings compared to busy hour reservation and conventional MPLS automatic bandwidth adjustment while keeping performance metrics of interest at the desired level. Consider how much gain we get applying the proposed algorithm.

Firstly, we compare performance of the proposed algorithm to that of the static busy hour resource reservation. The performance metric of interest is the amount of bandwidth required to transmit our traffic aggregates with a given overflow probability. Values of bandwidth allocation for our algorithm were computed using (11) such that the overflow probability is $\epsilon = 0.01$. Busy hour bandwidth allocation was chosen as the maximum bandwidth allocation computed by our algorithm. Note that it is different from the actual busy hour and represents the amount of bandwidth required to serve the traffic during the highest in control period of EWMA control chart. Parameter k of control charts was chosen such that ARL was always kept at 500. We also used the immediate update mechanism.

Figure 13 demonstrates bandwidth allocation according to considered algorithms. Estimated values of bandwidth allocation are shown in Table 1, where average values are shown. Note that our algorithm always provides significant performance gain compared to busy hour resource reserva-

tion. For trace 1 our algorithm allows to save 71% and 70% of bandwidth when EWMA change-point detection with $\gamma = 0.01$ and $\gamma = 0.001$ is used, respectively. For trace 2, these numbers are 70% and 69%, respectively.

Next we compare performance of the proposed algorithm to that of MPLS automatic bandwidth adjustment. Performance metrics of interest are the amount of LSP reestablishments and the amount of required bandwidth according to these two approaches. Values of bandwidth allocation were computed using (11) such that the overflow probability is $\epsilon = 0.01$. To highlight influence of k and γ on the performance of the proposed algorithm, parameter k was set such that ARL values for $\gamma = 0.01$ are always three times higher than those with $\gamma = 0.001$. For $\gamma = 0.01$, k was always chosen such that ARL is 300.

Figure 14 illustrates bandwidth allocation required by these two approaches. As one can notice, MPLS automatic bandwidth allocation always leads to higher required bandwidth for a given traffic aggregate. This is due to specific behavior of arrival statistics during sampling intervals over which the bandwidth allocation is estimated. This includes occasional outliers, drastic and gradual changes in arrival patterns. All these features lead to biased estimates of mean and variance that are further used to compute bandwidth allocation. Note that they also affect performance of the proposed algorithm.

It should also be stressed that the performance of MPLS bandwidth adjustment algorithm is highly sensitive to the choice of the sampling interval. When the sampling interval is too large, changes in arrival statistics are more likely to occur within a single interval. Large sampling intervals also lead to delays in LSP reestablishments with

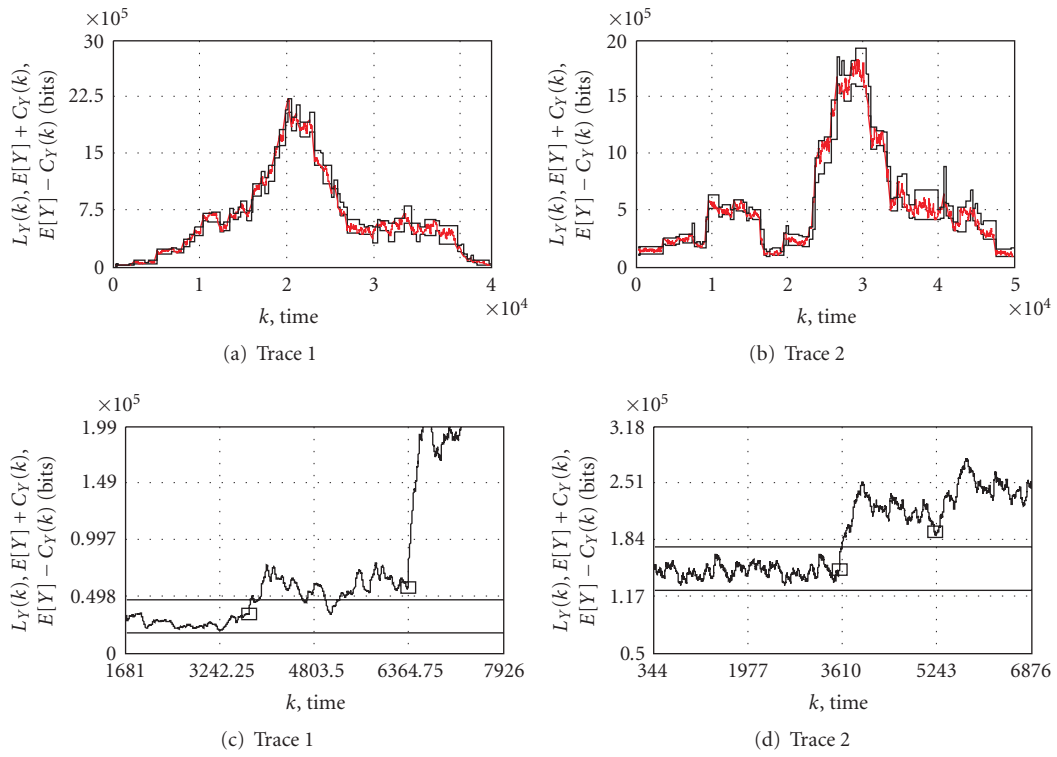


FIGURE 11: EWMA statistics: varying number of sources ($\gamma = 0.01$).

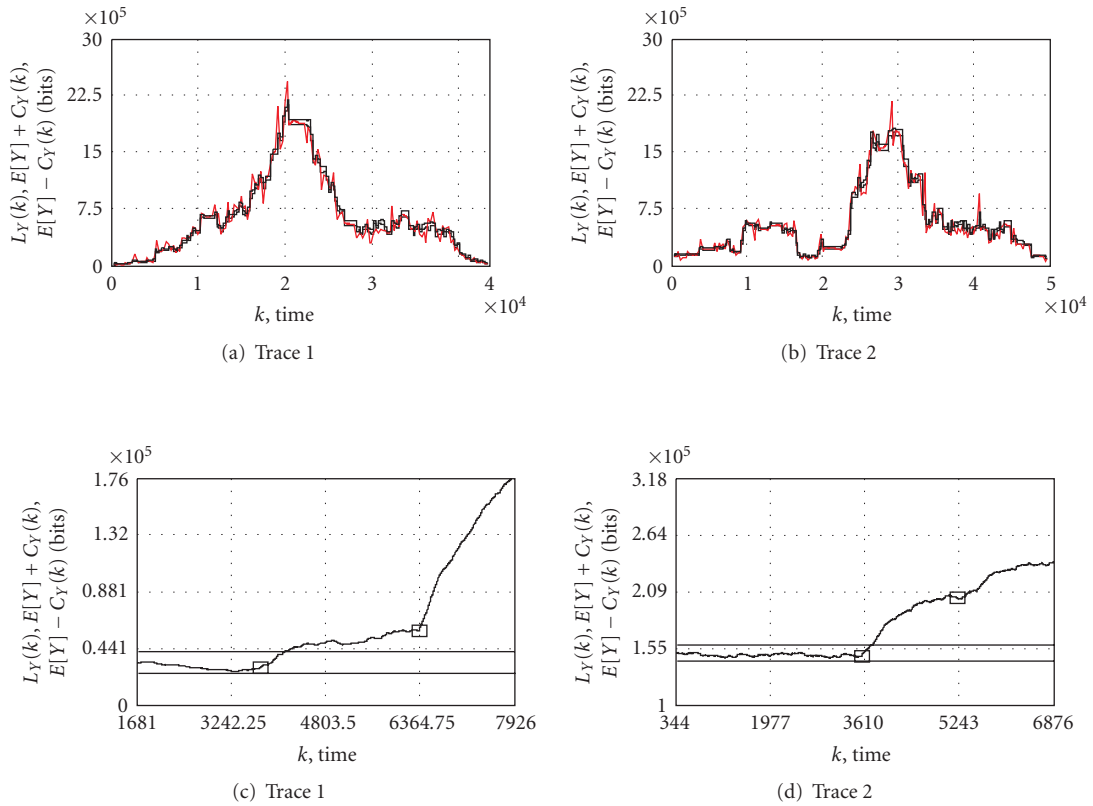


FIGURE 12: EWMA statistics: varying number of sources ($\gamma = 0.001$).

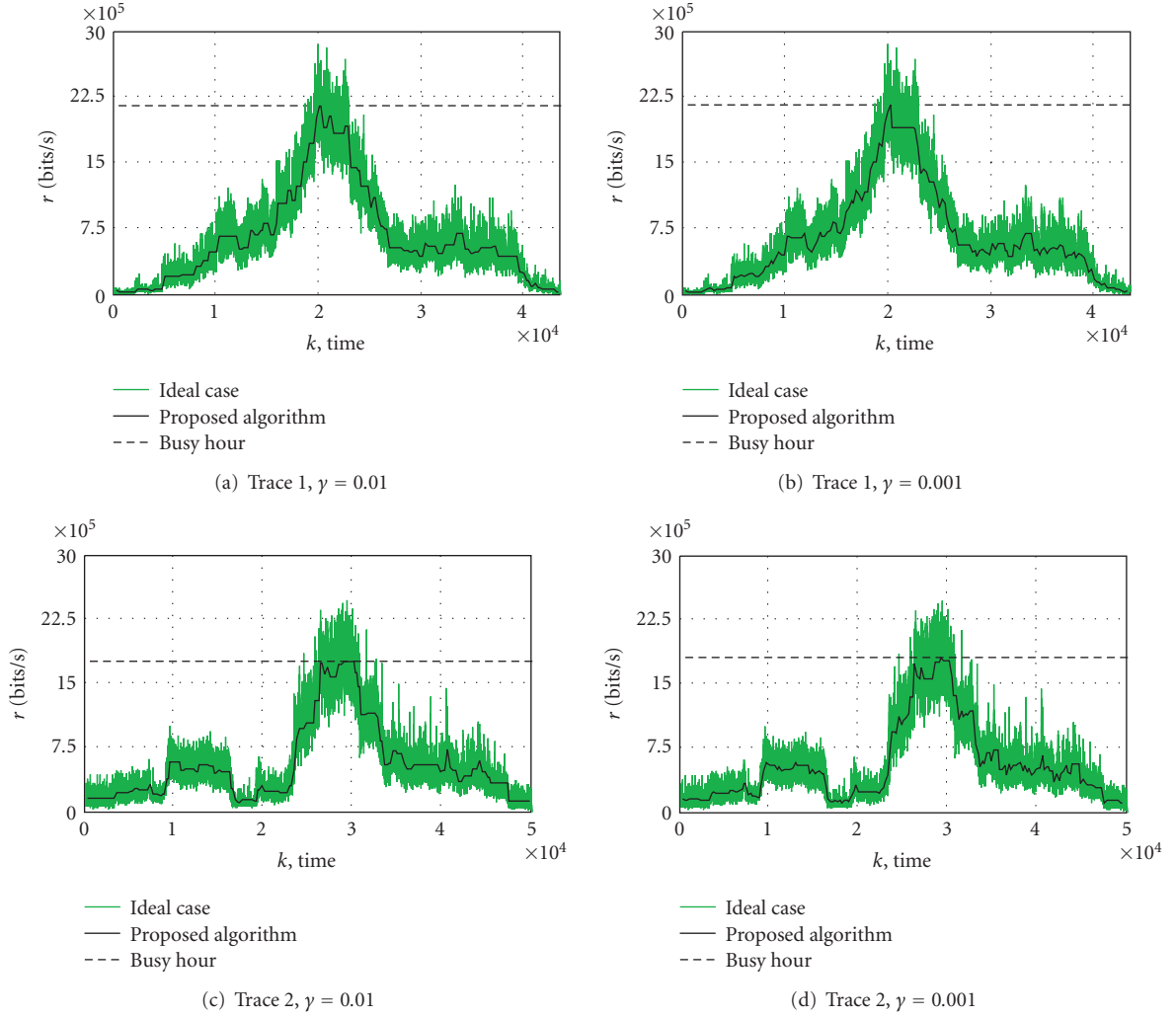


FIGURE 13: Bandwidth allocations for different approaches.

TABLE 1: Comparison with static busy hour bandwidth allocation.

Trace	Trace 1, bits/s.		Trace 2, bits/s.	
γ	$\gamma = 0.01$	$\gamma = 0.001$	$\gamma = 0.01$	$\gamma = 0.001$
Busy hour algorithm	2.14E6	2.16E6	1.76E6	1.80E6
Proposed approach	6.15E5	6.51E5	5.37E5	5.59E5

estimated bandwidth allocation. Due to presence of changes in arrival statistics, estimated bandwidth allocation may not be optimal at the end of intervals. On the other hand, when the sampling interval is too short, too many LSP reestablishments may burden the network with excessive signaling load.

Table 2 compares the amount of signaling load and average bandwidth required to transfer considered traffic aggregates for different lengths of the sampling interval and different values of γ . The average amount of bandwidth required to serve traffic aggregates according to our algorithm is considered as 100%. Parameters $\phi_{0.01}$ and $\phi_{0.001}$ denote the percentage of average bandwidth required

by MPLS automatic bandwidth adjustment compared to our algorithm. First of all, observing Tables 1 and 2 we note that MPLS automatic bandwidth adjustment allows to significantly decrease the amount of bandwidth required to serve considered traffic aggregates compared to static busy hour bandwidth allocation. Although the performance gain is significant, it has observable limit. Indeed, when we decreased the length of the sampling interval 10 times (from 500 to 50 seconds) the average bandwidth required to serve traffic aggregates decreased by only 0.04% for trace 1 and 0.05% for trace 2. This is very small performance gain, especially, considering that we are close to theoretical and practical limits. Indeed, further decrease in the length of the

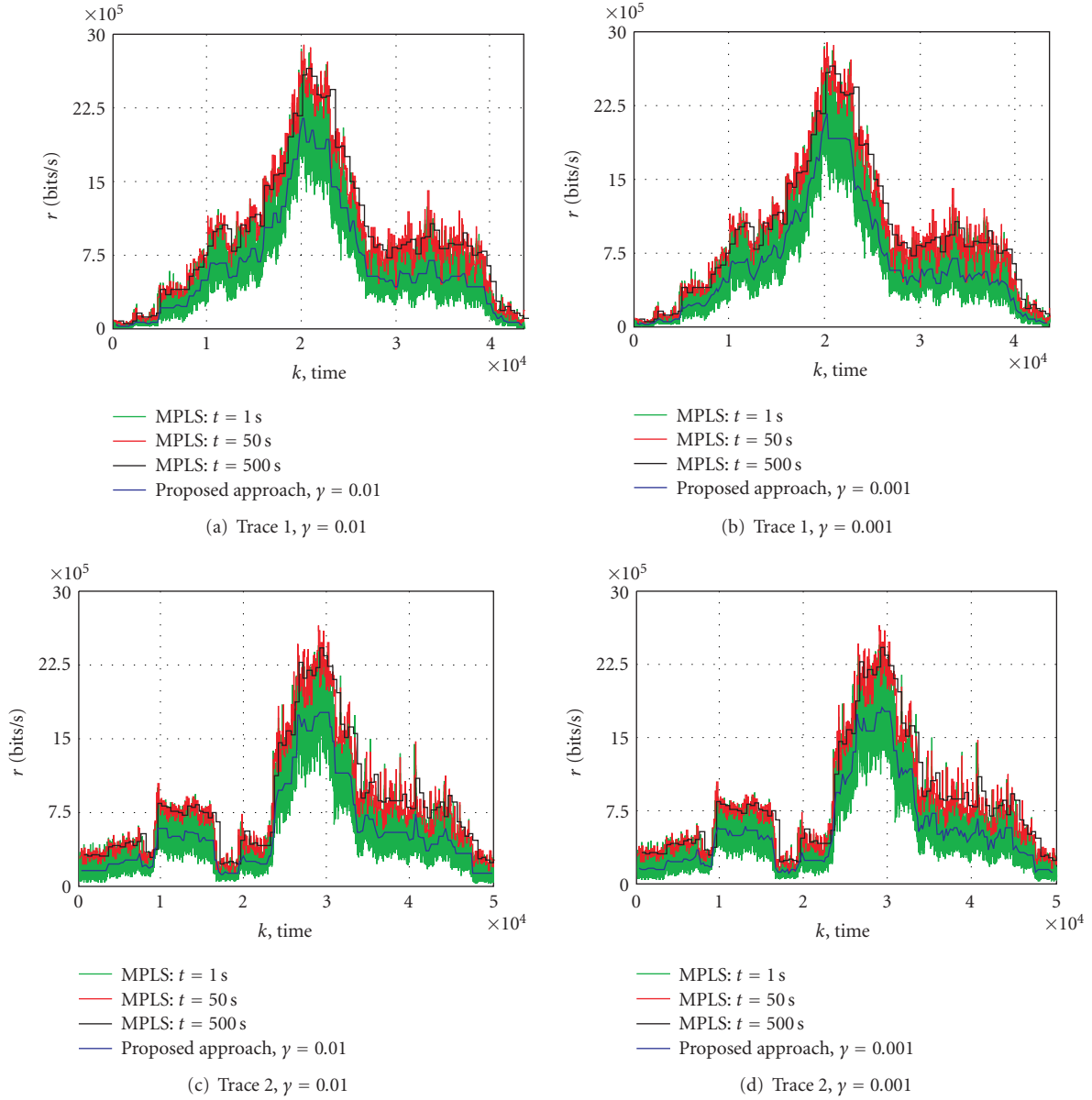


FIGURE 14: Bandwidth allocations according to different approaches.

sampling interval will lead to biased estimators of mean and variance and may overload the network with signaling load. On the other hand, as we observe, the proposed algorithm performs significantly better. For both values of γ it requires significantly less signaling load while still provides the same performance level in terms of losses. Noticeably that for $\gamma = 0.01$ our approach requires even less signaling than MPLS bandwidth adjustment with sampling interval set to 500 seconds.

Finally, influence of parameters k and γ on the performance of the proposed algorithm is also seen from Table 2. Indeed, three times lower ARL that was used with $\gamma = 0.001$ led to almost three times higher number of change detections. Although in our particular case it did not lead to extremely biased bandwidth allocation, it is still noticeable.

This effect is due to more frequent change detections in arriving traffic patterns leading to periods of uncertainty during which statistical parameters are estimated. During these periods we had to use the immediate update mechanism that may overestimate or underestimate the amount of required bandwidth.

It should be noted that, theoretically, the proposed algorithm cannot provide deterministic guarantees even if we use the immediate update mechanism and the target overflow probability is set to 0. The main reason is uncertainty introduced by warmup periods during which statistical characteristics of traffic aggregates are estimated. However, in all our experiments the cumulative length of warmup periods was significantly shorter compared to the overall length of in control periods.

TABLE 2: Comparison with MPLS automatic bandwidth adjustment.

Algorithm	Updates	r , bits/s.	$\phi_{0.01}$, %	$\phi_{0.001}$, %
Trace 1				
MPLS, $t = 50$ s.	870	9.21E5	150	141
MPLS, $t = 100$ s.	435	9.24E5	150	142
MPLS, $t = 200$ s.	145	9.35E5	152	143
MPLS, $t = 500$ s.	87	9.57E5	155	147
Proposal, $\gamma = 0.01$	66	6.15E5	100	100
Proposal, $\gamma = 0.001$	199	6.51E5	100	100
Trace 2				
MPLS, $t = 50$ s.	1001	8.13E5	151	145
MPLS, $t = 100$ s.	500	8.21E5	153	147
MPLS, $t = 200$ s.	166	8.31E5	155	149
MPLS, $t = 500$ s.	100	8.56E5	159	153
Proposal, $\gamma = 0.01$	75	5.37E5	100	100
Proposal, $\gamma = 0.001$	180	5.59E5	100	100

7. CONCLUSIONS

We proposed the dynamic resource description and reservation algorithm for QoS-aware networks. The core of the algorithm is statistical change detection procedure. We demonstrated that if the local mean of the traffic pattern changes in time, the required amount of resources needed to serve this traffic with given performance metrics can be appropriately adapted using already available features of MPLS framework. The only required change is EWMA change-detection algorithm that should be implemented in ingress MPLS routers. The proposed algorithm is well suited for non stationary type of the traffic whose statistical parameters change in time. Numerical examples demonstrated that when traffic patterns exhibit nonstationary behavior the proposed approach allows to save significant amount of resources compared to busy hour resource allocation and conventional MPLS automatic bandwidth adjustment.

We also note that content distribution services considered in this paper are just example of applications that may have time-varying traffic characteristics. In principle, the proposed approach can be used for any type of the aggregated traffic that experiences high variability due to time-varying statistical characteristics.

REFERENCES

- [1] K. Thompson, G. J. Miller, and R. Wilder, "Wide-area internet traffic patterns and characteristics," *IEEE Network*, vol. 11, no. 6, pp. 10–23, 1997.
- [2] S. Bates and S. McLaughlin, "Is VBR video non-stationary or self-similar? Implications for ATM traffic characterisation," in *Proceedings of the 10th European Simulation Multiconference (ESM '96)*, pp. 27–34, Budapest, Hungary, June 1996.
- [3] T. Karagiannis, M. Molle, M. Faloutsos, and A. Broido, "A nonstationary poisson view of Internet traffic," in *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '04)*, vol. 3, pp. 1558–1569, Hong Kong, March 2004.
- [4] J. Cao, W. Cleveland, D. Lin, and D. Sun, "On the non-stationarity of internet traffic," in *Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS '01)*, pp. 102–112, Cambridge, Mass, USA, June 2001.
- [5] J. Kilpi and I. Norros, "Testing the Gaussian approximation of aggregate traffic," in *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet Measurement (IMW '02)*, Marseille, France, November 2002, <http://www.imconf.net/imw-2002/proceedings.html>.
- [6] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson, "Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level," *IEEE/ACM Transactions on Networking*, vol. 5, no. 1, pp. 71–86, 1997.
- [7] A. Feldmann, A. Gilbert, P. Huang, and W. Willinger, "Dynamics of IP traffic: a study of the role of variability and the impact of control," in *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM '99)*, pp. 301–313, Cambridge, Mass, USA, August–September 1999.
- [8] W. Willinger, D. Alderson, and L. Li, "A pragmatic approach to dealing with high-variability in network measurements," in *Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement (IMC '04)*, pp. 88–100, Taormina, Sicily, Italy, October 2004.
- [9] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol label switching architecture," RFC 3031, IETF, 2001.
- [10] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. McManus, "Requirements for traffic engineering over MPLS," RFC 2702, IETF, 1999.
- [11] D. Katz, K. Kompella, and D. Yeung, "Traffic engineering (TE) extensions to OSPF version 2," RFC 3630, IETF, 2003.
- [12] H. Smit and T. Li, "IS-IS extensions for traffic engineering (TE)," RFC 3784, IETF, 2004.
- [13] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan, and G. Swallow, "RSVP-TE: extensions to RSVP for LSP tunnels," RFC 3209, IETF, 2001.
- [14] L. Andersson, P. Doolan, N. Feldman, A. Fredette, and B. Thomas, "LDP Specification," RFC 3036, IETF, 2001.

- [15] "MPEG-4 and H.263 video traces for network performance evaluation," Technical University of Berlin, <http://www.tkn.tu-berlin.de/research/trace/trace.html>, 2006.
- [16] "MPEG traffic archive," University of Wuerzburg, <http://www3.informatik.uni-wuerzburg.de/MPEG/traces/>, 2003.
- [17] G. Box and G. Jenkins, *Time Series Analysis: Forecasting and Control*, Holden Day, San Francisco, Calif, USA, 2nd edition, 1976.
- [18] S.-Q. Li and C.-L. Hwang, "Queue response to input correlation functions: discrete spectral analysis," *IEEE/ACM Transactions on Networking*, vol. 1, no. 5, pp. 522–533, 1993.
- [19] B. Hajek and L. He, "On variations of queue response for inputs with the same mean and autocorrelation function," *IEEE/ACM Transactions on Networking*, vol. 6, no. 5, pp. 588–598, 1998.
- [20] J. Wieringa, "Control charts for monitoring the mean of AR(1) data," Department of Econometrics, Faculty of Economic Sciences, University of Groningen, <http://www.ub.rug.nl/eldoc/som/a/98a09/98a09.pdf>, 2005.
- [21] J. Wieringa, "Statistical process control for serially correlated data," PhD Thesis, Department of Econometrics, Faculty of Economic Sciences, University of Groningen, Groningen, The Netherlands, 2005, <http://dissertations.ub.rug.nl/faculties/eco/1999/j.e.wieringa/>.
- [22] W. A. Shewhart, *Statistical Method from the Viewpoint of Quality Control*, Graduate School, Department of Agriculture, Washington, DC, USA, 1939.
- [23] E. S. Page, "A test for a change in a parameter occurring at an unknown point," *Biometrika*, vol. 42, no. 3-4, pp. 523–527, 1955.
- [24] Y. Koucheryavy, D. Moltchanov, and J. Harju, "A top-down approach to VoD traffic transmission over DiffServ domain using the AF PHB class," in *Proceedings of IEEE International Conference on Communications (ICC '03)*, vol. 1, pp. 243–249, Anchorage, Alaska, USA, May 2003.
- [25] R. G. Garroppo, S. Giordano, and M. Pagano, "Estimation of token bucket parameters for aggregated VoIP sources," *International Journal of Communication Systems*, vol. 15, no. 10, pp. 851–866, 2002.
- [26] 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, pp. 968–981, 1991.
- [27] F. Le Faucheur, "Protocol extensions for support of DiffServ-aware MPLS traffic engineering," RFC 4124, IETF, 2005.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

