Dynamic Resource Allocation Based on Online Traffic Prediction for Video Streams

Abdel-Karim Al-Tamimi
IEEE Student Member
Washington University in St. Louis
St. Louis, MO 63108 USA
aa7@cse.wustl.edu

Raj Jain
IEEE Fellow
Washington University in St. Louis
St. Louis, MO 63108 USA
jain@cse.wustl.edu

Chakchai So-In
IEEE Student Member
Washington University in St. Louis
St. Louis, MO 63108 USA
cs5@cse.wustl.edu

***Abstract*— In this paper, we propose a new dynamic resource allocation (DRA) scheme to support the constantly increasing online video stream traffic, especially high definition (HD) video streams. Our DRA scheme is based on online traffic prediction using seasonal time analysis. Our scheme seeks to provide accurate traffic prediction, to minimize the resource negotiation frequency, and to increase the utilization of the network resources while meeting maximum delay requirements. We validate our approach using various video traces, including our video collection of more than 50 HD video traces. We show through our results that our proposed scheme achieve up to 19.8% improvement in allocating bandwidth for short-length video traces, and up to 25% for long traces compared to the variable step-size adaptive (VSA) algorithm.**

Keywords- Bandwidth Prediction, Dynamic Resource Allocation, Multimedia over IP, QoS, VBR video, AVC, SAM Model, Seasonal ARIMA, VSA, QDBA, Video Traffic Prediction, Adaptive Multimedia, Admission Control.

I. INTRODUCTION

Video streaming traffic is continuously increasing its share of Internet traffic. Video streaming accounted for 27% of the Internet traffic in 2009, leveling up from 13% share during 2008 [8]. Video streams' traffic management is considered a challenging task because of their unique characteristics. Video streams are resource demanding traffic, and exhibit high variability in video frame sizes, which results in a bursty traffic. With the advent of a high definition (HD) video codec like MPEG-4 advanced video codec (AVC), the variability of the frame sizes have increased [1]. Online video traffic has high requirements for acceptable frame delay. Lost or delayed video frames are discarded at the end nodes if they do not arrive before their specified display time.

Video streams exhibit both short-term and long-term variances in frame size. For these reasons, static resource allocation is not preferable for managing video streams and providing quality of service (QoS) support. In order to conserve the computational resources and have a better control over the incoming video streams, per-flow management and bandwidth allocation is done usually at the edges of the network as shown in Fig. 1. Such position also allows the deployment of a better admission control mechanisms. In such schemes, the emphasis is to increase the network resources utilization while maintaining the desired level of QoS.

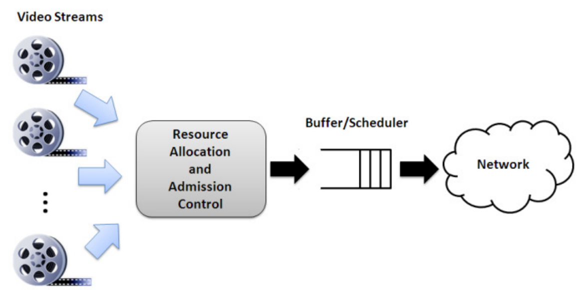

Figure 1. Dynamic resource allocation scheme

Dynamic resource allocation (DRA) schemes are especially important for live streams, where the video stream characteristics are not known in advance. In order to provide an accurate estimation of the needed network resources for a certain flow, which indicate the cost of transmitting such a flow, the chosen DRA scheme has to be able to predict the required bandwidth for future video frames. Such prediction is preferably dependent only on the information available from the incoming video stream (content-based), since the broadcasted information from the video source can either be unrepresentative of the video stream, or not available for live streams. To adjust the bandwidth assignment for a certain video stream, DRA renegotiate the assigned bandwidth for that flow.

The main goals for a DRA scheme are: to predict the longest possible period with the least prediction error, and to provide the best possible resource utilization with the lowest achievable frame delay. Content-based dynamic bandwidth allocation can be at the *video frame* level [2], *group of pictures* (GoP) level [3], or *scene* level [4, 5]. These types of QoS based video streaming management approaches are still considered essential for live video streaming [20, 21]. In [4] the authors proposed a DRA scheme based on neural network (NN) prediction. The scheme contains several modules to provide an accurate prediction. The results were based on 13175-frame video encoded via MPEG-1 VBR that is divided into 177 shots. To achieve the desired results, the first 50 shots were used as training samples. The proposed complicated approach reduces it applicability to support live video deadline requirements.

In [5], the authors proposed an object based video content classification scheme to map video scenes into their bandwidth resource requirements. This approach is also considerably complex. In [3], the author used a fixed-size adaptive least mean-square (LMS) error linear predictor to determine the required bandwidth allocation based on both a frame-level prediction, that requires separate prediction process for each frame type, and a simpler GoP level prediction. The used adaptive algorithm utilizes a fixed size adjustment to adapt to the traffic changes. In [7], the authors used the variable size LMS predictor proposed in [6] to predict the future bandwidth requirements based on the prediction of I-frames.

The main challenge with content-based predictions using a mathematical model is that they require precious computing resources and may not be applicable to video traces encoded with different encoding standards or settings [4]. Simplified Seasonal autoregressive integrated moving average (ARIMA) model, or SAM, has demonstrated its capability of modeling movies encoded with different encoding settings and standards [9, 10]. In addition, our simplification of SARIMA modeling as represented in SAM, as shown in Section II, allows real time implementation.

In this paper we propose a dynamic resource allocation scheme to support real time video streaming based on SAM model. In our design we consider the applicability of the scheme and compare it to the variable size LMS predictor using more than 54 HD video traces. In the next section, we describe SAM model and show our approach to predict future video frames. Section III demonstrates the design of our delay guaranteed SAM-based DRA scheme. Section IV describes our simulation experiments comparing SAM to variable size LMS predictor. Section V concludes this paper and provides a summary of our contributions.

II. SIMPLIFIED SEASONAL ARIMA MODEL (SAM)

In this section we discuss the SAM model and its representation as a seasonal autoregressive integrated moving-average (SARIMA) model. To allow a better representation of autoregressive (AR) model, we use the backward operator *B*, where:

$$B^j X_t = X_{t-j} \tag{1}$$

An autoregressive moving-average (ARMA) model that combines the both models can be expressed as:

$$x_t = \left(\varphi_1 B + \cdots + \varphi_p B^p\right) X_t + \varepsilon_t + \left(1 - \theta_1 B - \cdots - \theta_q B^q\right) \varepsilon_t \tag{2}$$

where ε_t ~N(0,σ^2) is the error term, and φ is the autoregressive coefficient, θ is the moving-average coefficient, and X_t is the frame size at time t. ARIMA can also be written as:

$$\varphi_p(B)\, X_t = \theta_q(B)\, \varepsilon_t \tag{3}$$

where

$$\varphi_p(B) = 1 - \varphi_1 B - \cdots - \varphi_p B^p \text{ and } \theta_q(B) = 1 - \theta_1 B - \cdots - \theta_q B^q$$

To remove the non-stationarity of a series, a differencing operator is introduced. This process results in an autoregressive integrated moving average (ARIMA) model. The order of differencing is denoted by *d*, and the differencing operator for *d* degree of differencing using backward operator notation is:

$$\nabla^d X_t = (1-B)^d X_t \tag{4}$$

and ARIMA process can be described as:

$$\varphi_p(B)\, \nabla^d X_t = \theta_q(B)\, \varepsilon_t \tag{5}$$

A seasonal time series is a series that exhibits a seasonal periodic behavior every *s* observations. This behavior can be expressed by extending the definition of ARIMA. Seasonal ARIMA or SARIMA has a seasonal autoregressive part of order *P* as Φ_P, and a seasonal moving average of *Q* order as Θ_Q, and a seasonal differencing of order *D* as ∇_s^D. SARIMA thus can be denoted as:

$$\varphi_p(B)\Phi_P(B^s)\nabla^d \nabla_s^D X_t = \theta_q(B)\Theta_Q(B^s)\varepsilon_t \tag{6}$$

A simpler notation to represent the order of each of the SARIMA model components is:

$$SARIMA = (p, d, q) \times (P, D, Q)^s \tag{7}$$

In order to represent a time series model using SARIMA, the model's components order and seasonality need first to be identified. The identification process requires human intervention to determine the best model to represent the analyzed data [12]. The next step is to estimate the model coefficients using different estimation methods. The most common used methods are: maximum likelihood (ML) estimation, conditional sum-of-squares (CSS) estimation, and a hybrid approach where the starting values are estimated using CSS then ML is used to complete the estimation process (CSS-ML) [12]. This two-step process, (i.e. determining the model order and estimating its parameters), is a time consuming process and requires a substantial statistical background to identify the best possible model to represent a time series [13].

A. The SAM Model

SAM provides the means to model video traces accurately without the need of human intervention. In previous research results, it was shown that SAM is capable of capturing the statistical characteristics of video traces with less than 1% difference from the optimal models for each video trace. The model has been tested against video traces encoded using different encoding settings and standards: MPEG-Part2, MPEG4-Part10/AVC, and AVC's scalable extension for temporal scalability (SVC-TS) [9, 10]. SAM can be represented using the simplified notation:

$$SAM = (1,0,1) \times (1,1,1)^z \tag{8}$$

where *z* here represents the seasonality of the video trace. This means that SAM requires only 4 coefficients to be estimated. These coefficients are: AR coefficient (φ), MA coefficient (θ), seasonal AR or SAR coefficient (Φs), and seasonal MA or SMA coefficient (Θs). SAM can be represented using SARIMA notation as:

$$\varphi(B)\Phi(B^s)\nabla_s X_t = \theta(B)\Theta(B^s)\,\varepsilon_t \text{, or}$$

$$(1-\Phi_s B^s)(1-\varphi B)(1-B^s)\, X_t = (1-\theta B)(1-\Theta_s B^s)\, \varepsilon_t$$

or we can simplify this further in *difference equation* form:

$$\begin{aligned} X_t = {} & X_{t-1} + \varphi X_{t-1} - \varphi X_{t-2} + \Phi_s X_{t-s} - \varphi\Phi_s X_{t-s-1} - \Phi_s X_{t-s-1} \\ & + \varphi\Phi_s X_{t-s-2} - \theta\varepsilon_{t-1} - \Theta_s \varepsilon_{t-s} + \theta\Theta_s \varepsilon_{t-s-1} + \varepsilon_t \end{aligned} \tag{9}$$

Since SAM requires only 4 parameters to be estimated, such convenient simplification allows it to be used in real time applications.

B. Model Parameters Estimation Methods

The estimations of SAM parameters can be achieved, as mentioned before, using ML, CSS, or CSS-ML methods. Most literature books suggest ML as the best option to obtain the best parameters estimations [12, 13].

To support real time prediction, the estimation method should provide a good tradeoff between high accuracy and high computation speed. To determine the best suitable estimation method, we compared ML, CSS, and CSS-ML methods in modeling our collection of 54 HD video traces encoded with AVC codec. The videos cover a wide variety of texture/details and motion levels. The selected video traces represent a random set of the most visited videos, and three random videos from each of the 15 video subcategories available to YouTube's users [11]. The average video trace length is 3212 frames, with a maximum of 9388 frames, and a minimum of 580 frames. The encoding settings have been chosen to confirm with the expert's recommendations and the majority of HD YouTube videos [14, 23, 24].

The results of the comparison are shown in Table 2 and Table 3. First we compared the three methods in the total time needed to model our collection of video traces. As shown in Table 2 CSS method has a clear advantage over both ML and CSS-ML. Using CSS it took only 0.22 seconds on average to model a full video trace, compared to 39.54 seconds using ML.

As mentioned before, high computation speed should not come at the expense of modeling accuracy. We used three statistical measures to compare the modeling accuracy of the three methods: Mean Absolute Error (MAE), Mean Absolute Relative Error (MARE), and Root Mean Square Error (RMSE):

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|e_i| \quad , \quad MARE = \frac{1}{N}\sum_{i=1}^{N}\frac{|e_i|}{X_i}$$

$$and \quad RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(e_i)^2}$$

where N is number of video frames, e_i is the prediction error at the i-th frame, and X_i is the i-th frame size.

Both Table I and Table II show that the difference between ML and CSS, in terms of accuracy, is less than 2.5%. We argue that such degradation of accuracy is acceptable compared to the significant boost in computation speed. Based on these results we recommend using CSS as the parameter estimation method for SAM. In this paper, all our results are based on using CSS method.

TABLE I. ESTIMATION METHODS COMPARISON RESULTS

Comparison/Method	**ML**	**CSS**	**CSS-ML**
Total execution time (s)	2135.06	15.6	1202.68
Average time per video (s)	39.54	0.22	22.27
MAE (average)	12437.8	12568.1	12523.48
MARE (average)	2.395	2.447	2.417
RMSE (average)	21712.3	22162.3	22004.9

TABLE II. PERCENTAGE OF IMPROVEMENT BETWEEN ESTIMATION METHODS

Comparison/Method	**ML vs CSS**	**CSS-ML vs CSS**	**ML vs CSS-ML**
MAE	1.04%	0.35%	0.68%
MARE	2.12%	1.27%	0.86%
RMSE	2.03%	0.71%	1.33%

C. Forecasting Using SAM

In [14], the authors showed that SAM requires only around 100 previous data values to provide an accurate representative model. This observation confirms with the recommended guidelines for forecasting using ARIMA models [13]. In addition, in [4], the authors showed that most useful traffic information are presented in the short-term bandwidth statistics. This approach provides a valid method to achieve the desired forecasting without sacrificing performance since in practical applications the resulting forecasts are dependent significantly only on the recent values of the observed data series [15].

We achieve forecasting or prediction of future values of X_t directly from the previously mentioned SAM's difference equation. In this process the values of X_t and the estimation error terms are substituted as follows:

$$[X_{t+i}] = \begin{cases} X_{t+i} & i \le 0 \\ \hat{X}_t(i) & i > 0 \end{cases} \quad and \ [\varepsilon_{t+i}] = \begin{cases} \varepsilon_{t+i} & i \le 0 \\ 0 & i > 0 \end{cases} \qquad (10.1, 10.2)$$

where $\hat{X}_t$ is the estimated frame size at time *t*. Such assumptions are valid since forecasts values are unaffected by the small changes that are introduced by the estimation errors [15].

III. SAM-BASED DRA SCHEME

In this section, we discuss our approach to determine the required adjustment to the allocated bandwidth upon detecting a trend change. In our scheme, SAM predictor analyzes the observed traffic information and then it accordingly predicts the future incoming traffic and adjusts the previously allocated bandwidth. In case of a prediction error, if the traffic predictor allocates more bandwidth for the incoming flow than it requires, there will be a waste of network resources (link utilization). On the other hand, if the prediction process results in a less bandwidth than the flow requires, the difference should be buffered and then sent later within the acceptable delay limits for live video streams

Fig. 2 shows the proposed SAM based DRA model. The incoming video flow is processed through the SAM-based stream resource controller (SRC), where the prediction process is performed. The prediction difference due to the prediction errors is buffered. At renegotiation points, a simple request/response mechanism is used to communicate with the network resource manager (NRM). Depending on the available network resources, NRM determines whether the requested increase or a new flow request can be supported. In case the incoming flow cannot be supported by the network, SRC may send a feedback to the video source encoder to use a lower bit rate [17]. To simplify the simulation, we assume that all bandwidth requests are granted.

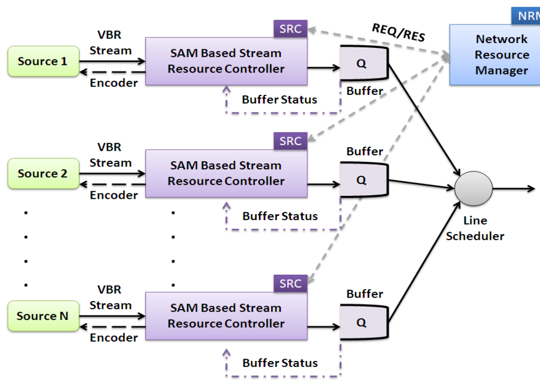

Figure 2. SAM-based DRA model

As can be noticed from Fig. 2, by using SAM to predict and reserve the bandwidth dynamically, the allocation problem has changed from supporting highly variant incoming video flows to servicing the predicted allocations and buffering the possible prediction errors. Thus, the better the predictor, the better the system performance in servicing the existing flows and admitting new video flows.

IV. EXPERIMENTAL RESULTS

SAM-based SRC aims to manage the dynamic bandwidth allocation while meeting the one-way delay requirement. IPTV QoS requirements for MPEG-4 AVC encoded HDTV service, as described in DSL Forum technical report (TR-126) for triple play service [25], are set to maximum of 200 ms one-way delay. To meet the desired maximum deadline, we incorporated in our DRA design the QoS guaranteed dynamic bandwidth allocation (QDBA) algorithm [18]. QDBA operates at the GoP-level. This approach provides two advantages to the forecasting process: it simplifies the calculation of the incoming flows, and it also acts as a smoother for the variable video stream allowing easier prediction.

QDBA algorithm compares the required bandwidth allocation at each time slot, taking into consideration the buffer status, against both the predicted and currently allocated bandwidth. If the current rate is higher than the required rate considering the required link utilization, then it renegotiates a lower rate that is the max of the predicted and required rates. If the allocated rate is lower than the required rate, the renegotiated rate is set to the required rate. Otherwise, the allocated rate remains the same.

We compare the proposed SAM-based traffic predictor to the traffic predictor based on the non-linear variable step-size adaptive (VSA) algorithm proposed in [7]. VSA is an improvement over the fixed step-size adaptive (FSA) algorithm [22, 3]. VSA bases its prediction on the GoP size instead of the frame size. This approach was favored since modeling and predicting the GoP size is considered a simpler problem. By operating on the frame-level instead, SAM aims to provide better prediction results. Let us consider G_i the i-th GoP size, then the p-th order of VSA predictor is:

$$\hat{G}_{i+1} = \sum_{j=0}^{p-1} w_j G_{i-1} = W^T G_i \tag{11}$$

where $\hat{G}_{i+1}$ is the forecasted size of the next GoP, p is the order of the predictor, w_j are the prediction filter coefficients for $(j = 0,1,\ldots, p-1)$. The predictor's order is chosen empirically to achieve the best results. The prediction error and filter coefficients can be calculated using:

$$e_i = G_i - \hat{G}_i \text{ and } W_{i+1} = W_i + \frac{\mu_i e_i G_i}{\|G_i\|^2} \qquad (12.1,\ 12.2)$$

where $\|G_i\|^2 = G_i^T \times G_i$. Instead of using a fixed value for the updating coefficient μ, its value is updated to allow variable step-size adjustments. Increasing the value of μ results in fast convergence but at the expense of higher prediction errors. Smaller μ value results in smaller prediction errors with slower convergence rate. For highly variable input stream like HD video traces, it is important to choose the correct value for μ to allow fast adaption to the stream variation with the lowest possible prediction errors. μ_i is updated using:

$$\mu'_{i+1} = \alpha\mu_k + \gamma(q_1 e_i^2 + q_2 e_{i-1}^2) \tag{13.1}$$

$$\mu_{i+1} = \begin{cases} \mu_{\max} & \text{if } \mu'_{i+1} > \mu_{\max} \\ \mu_{\min} & \text{if } \mu'_{i+1} < \mu_{\min} \\ \mu'_{i+1} & \text{otherwise} \end{cases} \tag{13.2}$$

here α is the previous μ weight, γ is the collective error terms weight, q_1 and q_2 are the current and the previous prediction errors weights respectively. $\mu_{\max}$ and $\mu_{\min}$ are used to bound the step-size adjustment. $\mu_{\max}$ is chosen to ensure that the mean square error (MSE) remains bounded, while $\mu_{\min}$ is the same value as that chosen for FSA algorithm. As shown here, VSA requires 7 coefficients to be determined empirically before being deployed in real-time applications, which is considered a down-side to this approach. SAM, on the other hand, does not require any prior information or prior empirical calculations.

Following the suggestions in [18], we chose α = 0.98, γ =0.015, q_1 = 0.7, and q_2 = 0.3. Following the suggestions in [7], we set the initial value of the updating coefficient μ_0 = $\mu_{\min}$ = 0.009, $\mu_{\max}$ = 0.3, and the prediction order p = 12. In our simulations we found out that to achieve the best prediction results we need to set $\mu_{\max}$ = 0.03, and p = 5. In our experiments we used a quad-core i7 (2.8Ghz) machine with 6GB of RAM.

As we stated before, one of the main targets in bandwidth allocation schemes design is to minimize the number of the renegotiations points. Therefore, we modified VSA to allow the prediction for more than one GoP. The modification simply allows VSA to operate on aggregation of multiple GoP sizes, instead of one GoP size.

The performance of the two compared dynamic bandwidth predictors using QDBA algorithm is measured using three parameters: renegotiation frequency, the total allocated bandwidth, and the total buffer usage or occupancy. A better predictor will result in fewer prediction errors to be buffered (smaller queue occupancy), better future prediction (fewer renegotiation points), and better utilization of the network resources under the defined delay requirements (lower bandwidth allocation rate).

Table III shows the average performance comparison results using our collection of 54 HD video traces for maximum allowed delay of 100 ms, and required bandwidth utilization of $\rho = 0.9$. SAM-n /VSA-n indicates the performing SAM/VSA over n-aggregated GoPs. For example, SAM-2 means two GoP sizes are aggregated.

TABLE III. PERCENTAGE OF IMPROVEMENT FOR USING SAM OVER VSA

Comparison/Method	SAM-1 Vs VSA-1	SAM-2 Vs VSA-2	SAM-3 Vs VSA-3
Allocated Bandwidth	19.8%	8.6%	7.7%
Negotiation Freq.	0.5%	3.5%	5.77%
Queue Occupancy	25.2%	13.8%	13%

We notice that SAM outperforms VSA in all the performance comparisons due to its higher ability to predict future traffic. Even with the low number of frames tested with average of ~3000 frames, SAM provides 7.7% (SAM-3) to 19.8% (SAM-1) bandwidth utilization improvement, and 13% (SAM-1) to 25.2% (SAM-1) queue occupancy improvement. By increasing the number of aggregated GoPs the difference between the two approaches becomes lower because it represents a smoother version of the video trace. The same observation is noticed in the queue occupancy comparison. It is important to mention that increasing data aggregation results in higher error rates and thus higher queue occupancy and higher bandwidth allocation rates. Table IV shows the average percentage of increment for both queue occupancy and allocated bandwidth.

TABLE IV. PERCENTAGE OF INCREMENT FOR PERFORMANCE METRICS

Comparison	GoP-2 vs GoP-1	GoP-3 vs GoP-1
Allocated Bandwidth	12%	14.8%
Queue Occupancy	10.6%	12.8%

The improvements of negotiation frequency increases with increasing the number of aggregated GoPs since the total number of renegotiations are fewer with higher level of aggregation. For instance, 0.5% reflects the improvement from 105 (VSA) to 104 (SAM), where 3.5% reflects the improvement from 55 (VSA) to 53 (SAM). Renegotiation frequency is almost cut in half when using GoP-2 aggregation. Thus, as a tradeoff between high accuracy and lower renegotiation frequency GoP-2 aggregation can be used.

Fig. 3 illustrates a close-up comparison between the VSA-1 and SAM-1 in their ability to predict future traffic using one of the simulated video trace samples. It is clear that SAM predicts the traffic better especially in the case of sudden events since it operates on frames level.

To support our assumption that the difference between SAM and VSA will be even more substantial using longer video traces, we compared the two methods using three long traces obtained from [19]. The three selected traces are chosen to represent various video characteristics: *Silence of the Lambs* (~30 min, 52384 frames), *Tokyo Olympics* (~74 min, 131520 frames), and *Star Wars IV* (~30 min, 52384 frames), representing action, thriller, and sports video genres, respectively. All these movies are encoded using AVC main profile, with a frame rate of 30 fps, a GoP size of 16, 7 B-frames per GoP, and a quantization level of 28 for all frames.

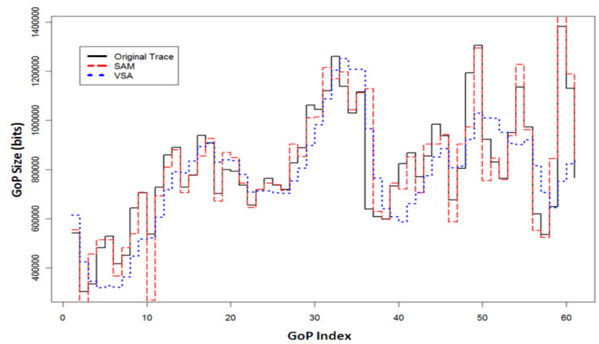

Figure 3. SAM versus VSA prediction rate comparison

Table V shows the performance analysis results for the three movies tested movie traces. Again, SAM has a clear advantage over VSA in all the tested GoP aggregations. We can also see that the performance gain has increased using longer traces. Therefore, using SAM especially for live and continuous streams applications like IPTV will result in better utilization of network resources.

TABLE V. PERCENTAGE OF IMPROVEMENT FOR SAM OVER VSA

Video Trace	Comparison/Method	SAM-1 Vs VSA-1	SAM-2 Vs VSA-2	SAM-3 Vs VSA-3
Star Wars IV	*Allocated Bandwidth*	21.7%	25.3%	17.7%
	Negotiation Freq.	1.68%	5.8%	7.95%
	Queue Occupancy	22.8%	26.6%	19.3%
Silence of the Lambs	*Allocated Bandwidth*	22.4%	15.5%	22.6%
	Negotiation Freq.	3.9%	5.65%	8.04%
	Queue Occupancy	23%	16%	23.8%
Tokyo Olympics	*Allocated Bandwidth*	25%	24.7%	25.8%
	Negotiation Freq.	3.1%	8.7%	9.4%
	Queue Occupancy	27.9%	27.2%	28.2%

As we stated before, our proposed design takes into consideration the maximum allowed bandwidth delay for the incoming video flows. Using QDBA algorithm enforces the acceptable deadline delay and provides a good supports for the QoS requirements. Fig. 4 shows how the deadline requirement is met when the maximum delay requirement is set to T = 200 ms, and the required bandwidth utilization is set to $\rho = 0.9$ for *Silence of the Lambs* video trace.

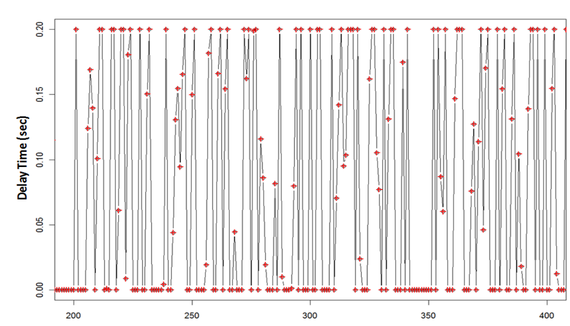

Figure 4. Meeting delay requirements in SAM-based DRA (T=200ms)

As can be noticed, the proposed SAM-based DRA scheme meets the deadline requirements while maximizing the utilization of the available network resources through providing better prediction performance.

V. Conclusions and Discussions

In this paper we proposed a dynamic resource allocation scheme based on SAM video trace model to provide a better support for online video streams. Our contributions through this paper can be summarized in the following points:

- We illustrated the mechanism of using SAM model to forecast the incoming video frames depending only on the short-term history of the previously observed frames. Additionally, we compared the SAM modeling accuracy using three parameter-estimation algorithms (CSS, ML, and CSS-ML) to achieve a higher computational performance. Using CSS algorithm provides a significant boost in computation speed, 0.22 seconds per video versus 39.54 using ML on average, with less than 2.5% loss of accuracy in our comparisons using MAE, MARE, and RMSE.

- Based on our proposed DRA scheme, we compared SAM and VSA using our collection of 54 HD video traces. Based on the three performance measures: allocated bandwidth, renegotiation frequency, and queue occupancy, we showed that SAM outperforms VSA in all of them providing up to 7.7% (SAM-3) to 19.8% (SAM-1) improvement in bandwidth utilization, and up to 13% (SAM-1) to 25.2% (SAM-1) improvement in queue occupancy on average.

- We extended our analysis results by comparing VSA and SAM using three long video traces representing different video genres. The results confirmed our assumption that SAM has a clear edge over VSA especially for long video traces. Thus, we believe that SAM will provide significant network resource utilization improvements for continuous video stream applications like IPTV.

The dynamic bandwidth allocation performance improvement using our proposed SAM-based DRA scheme and its proven real time applicability, while meeting the strict delay deadlines of live videos, makes it a strong candidate for real time deployment to support live video streams, especially for continuous video stream applications like IPTV, and improve network resources utilization.

VI. References

[1] Van der Auwera, G., et al. 2008. Traffic and quality characterization of the H.264/AVC scalable video coding extension. Adv. MultiMedia 2008, 2 (Jan. 2008).

[2] Choi, J.-Y., Shin, J. 2005. Content-Aware Packet-Level Interleaving Method for Video Transmission over Wireless Networks. Lecture Note in Computer Science. 2005, ISSU 3510, pages 149-158.

[3] Adas, A. M. 1998. Using adaptive linear prediction to support real-time VBR video under RCBR network service model. IEEE/ACM Trans. Netw. 6, 5 (Oct. 1998), 635-644.

[4] Wu, M., et al. 2001. Dynamic resource allocation via video content and short-term traffic statistics, IEEE Transactions on Multimedia, vol. 3, no. 2, pp. 186-199, 2001.

[5] Bocheck, P. and Chang, S.-F. 1998. Content-based VBR traffic modeling and its application to dynamic network resource allocation , Res. Rep. 48c-98-20, Columbia University, New York, NY, USA, 1998.

[6] Kwong ,R. H. and Johnston, E.W. 1992. A variable step size LMS algorithm, IEEE Trans. Signal Processing, vol. 40, pp. 1633–1642, July 1992.

[7] Zhao, H., et al. 2002. A Fast Non-Linear Adaptive Algorithm for Video Traffic Prediction. In Proc. of the international. Conf. on information Tech. (April 08-10, 2002).

[8] Albrecht , C., "Survey: Online Video Up to 27% of Internet Traffic", (online) [http://tinyurl.com/yzpzoew] (Mar 2010).

[9] Al-Tamimi, A., et al. Modeling and generation of AVC and SVC-TS mobile video traces for broadband access networks. In Proc. of the 1st ACM SIGMM Conference on Multimedia Sys. (Phoenix, Arizona, USA, Feb. 22-23). MMSys '10.

[10] Al-Tamimi, A., et al. Modeling and Resource Allocation for Mobile Video over WiMAX Broadband Wireless Networks," IEEE Journal on Selected Areas in Comm., Special Issue on Wireless Video Tran., vol. 28, no. 3, pp. 354-365, April 2010

[11] The Top 5 video streaming websites 2010. (online) URL=[http://www.techsupportalert.com/top-5-video-streamingwebsites.htm], (March 2010).

[12] Chatfield, C. 2003. The Analysis of Time Series: An Introduction, Sixth Edition, Chapman & Hall/CRC 2003.

[13] Montgomery, D.C., et al. 1990. Forecasting and Time Series Analysis, 2nd Ed., New York, McGraw-Hill, 1990

[14] Al-Tamimi, A.,et al. 2010c. Modeling and Prediction of High Definition Video Traffic: A Real-World Case Study. In Proc. on Advances in Multimedia (MMEDIA), June 2010, in press.

[15] Box, G. E. P., et al. 2008. Time Series Analysis: Forecasting and Control. Wiley Series in Probability and Statistics. 4th Edition. ISBN-10: 0470272848.

[16] Melamed, B. and Pendarakis, D. E. 1998. Modeling full-length VBR video using markov-renewal-modulated TES models. IEEE J. Select. Areas Commun., vol. 16, pp. 600–611, June 1998.

[17] Su, G.-M. and Wu, M. Efficient bandwidth resource allocation for low-delay multiuser MPEG-4 video transmission. In IEEE Inter. Conf. on Comm., vol. 3, ppS. 1308-1312, 2004.

[18] Zhao, H., et al. 2004. Delay guaranteed bandwidth allocation for real-time video delivery. IEEE Proc. Commun., 2004, 151, (6), pp. 553–558.

[19] Mobile video traces, Mobile Devices: Video Traces, Arizona State University (ASU) and Aalborg University (AAU) URL=[http://mobiledevices.kom.aau.dk/research/traffic and channel measurements/video traces/]. (April 2010).

[20] Cisco Tech Notes. "Implementing QoS Solutions for H.323 Video Conferencing over IP," Document ID = 21662. Feb 2008. URL=[http://www.cisco.com/en/US/tech/tk543/tk757/technologies_tech_note09186a0080094968.shtml]. (April 2010).

[21] Y. Bai, and M. R. Ito, "Managing QoS requirements for video streaming: from intra-node to inter-node: Research Articles," International J. Commun. Syst. 19, 5 (Jun. 2006), 545-565. DOI= http://dx.doi.org/10.1002/dac.v19:5.

[22] B. Widrow, "Adaptive Signal Processing," Prentice-Hall, Englewood Cliffs, N.J. 1985.

[23] A. Al-Tamimi, R. Jain, C. So-in, "Statistical Analysis and Modeling of High Definition Video Traces", International Conference on Multimedia and Expo, ICME 2010, in press.

[24] J. Ozer, "Producing H.264 Video for Flash: An Overview" ,URL=http://www.streaminglearningcenter.com/articles/41/1/Producing-H264-Video-for-Flash-An-Overview/Page1.html.

[25] DSL Forum Tech. Rep, "Triple-play Service Quality of Experience (QoE) Requirements," TR-126. (online) URL=[http://www.broadband-forum.org/technical/download/TR-126.pdf] (April 2010).