Risk Management for Video-on-Demand Servers leveraging Demand Forecast *

Di Niu, Hong Xu, Baochun Li
Department of Electrical and Computer Engineering
University of Toronto
{dniu, henryxu, bli}@eecg.toronto.edu

Shuqiao Zhao
Multimedia Development Group
UUSee, Inc.
shuqiao.zhao@gmail.com

ABSTRACT

Video-on-demand (VoD) servers are usually over-provisioned for peak demands, incurring a low average resource efficiency. However, bandwidth shortage may still occur for individual videos as they share and contend for server resources. In this position paper, we propose a predictive workload management system for VoD servers targeting bandwidth. The system draws belief about future demand as well as demand volatility based on demand history using time series forecasting techniques. The prediction enables dynamic and efficient server bandwidth reservation with QoS guarantees. More importantly, we use a hedging technique similar to investment portfolio management and distribute workloads to multiple servers exploiting demand anti-correlation. The proposed system consolidates the workloads, enhances resource utilization, while in the meantime effectively controlling risk of server overload. The proposed methods are evaluated based on real-world VoD traces.

Categories and Subject Descriptors

C.2.3 [**Network Operations**]: Network Management; Network Monitoring; C.4 [**Performance of Systems**]: Modeling Techniques

General Terms

Algorithms, Measurement, Performance, Reliability

Keywords

Video-on-Demand, Risk Management, Hedging, Demand Prediction, Bandwidth Reservation

1. INTRODUCTION

A large-scale Video on Demand (VoD) system involves millions of users watching movies, TV episodes and other videos streamed from a huge library of video channels. As a user must download at a rate no smaller than the video playback rate to watch a video smoothly, bandwidth constitutes the performance bottleneck in these systems. VoD servers therefore require abundant egress bandwidth as well as disk I/O speed to read and deliver video continuously.

VoD server capacity is usually overly provisioned for the anticipated peak demand, incurring low utilization during non-peak hours. Despite over-provision, bandwidth shortage can still occur for an individual video, if the instantaneous demand for it exceeds the aggregate capacity of the servers hosting the video. Undoubtedly, in the presence of demand fluctuation and uncertainty, one challenge facing all VoD services is to economically plan the capacity, minimizing unnecessary over-provision costs while in the meantime controlling risk of under-provision or server overload.

Dynamic *bandwidth reservation*, if realized, can enhance resource efficiency and control quality at the same time, because now we can limit the server bandwidth dedicated for each video channel, allowing efficient and safe use of the remaining idle bandwidth by other channels or applications. Specifically, if the bandwidth reserved for a channel exceeds the demand for it, other channels/applications can use the unreserved bandwidth or the idle part in the reserved bandwidth. On the other hand, if the demand for the channel exceeds the reserved bandwidth, the aggregate downloading rate of the video will be capped by the reserved bandwidth. The above bandwidth reservation function can be readily implemented with software rate limiting at network interfaces of servers.

Apparently, there is a tradeoff between QoS and resource efficiency: when more bandwidth is reserved, the risk of shortage is lowered, yet the unreserved bandwidth that can be promised for other purposes is reduced undesirably. Therefore, an important research problem is to judiciously decide the right amount of server bandwidth to be reserved for each channel as demand evolves, with two major challenges. First, bandwidth reservation must be predictive instead of reactive, since when bandwidth shortage occurs in a channel, a delay is needed to replicate the video to other servers in order to increase the capacity available for this video. With the presence of configuration and replication delays, a reactive scheme is too slow to match demand changes closely. Second, since video servers are usually distributed, decisions have to made regarding how to split the workload among multiple servers as well as where to place the content.

In this paper, from a novel perspective, we view the bandwidth demand for each video in short-term future as a random variable with predictable mean and variance, just as

*Area chair: Reza Rejaie

MM'11, November 28–December 1, 2011, Scottsdale, Arizona, USA.

investments in financial markets are assumed to have an expected return subject to random risks. We propose to book bandwidth for video channels that accommodates not only their projected mean demands but also the demand variation. We formulate a mean-variance optimization problem through which each server consolidates workloads based on demand anti-correlation, by maximizing the expected demand it can serve while confining the probability of server overload to a small threshold. Such a formulation is inspired by investment diversification theory [**?**], according to which, a stock investor chooses a diversified portfolio of stocks to minimize her return uncertainty given a certain expected return level. However, we modify such a portfolio theory to adapt to our scenario of video server management. Before introducing our workload portfolio management framework, we also briefly mention how to make prediction about demand statistics using seasonal ARIMA models [**?**] and GARCH models [**?**], widely used in econometrics.

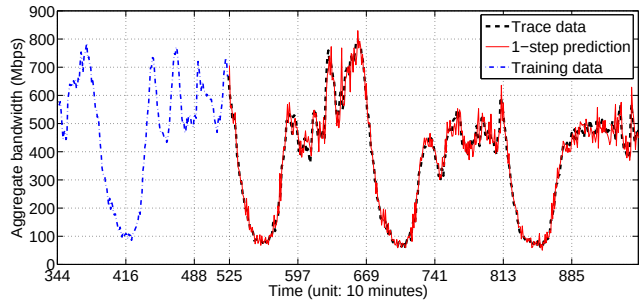
The resulted workload management system makes decisions with respect to bandwidth reservation, content placement, and load direction, which are updated at 10-minute frequency and readily implementable in real-world systems. Trace-driven simulations show that, our algorithm can enhance utilization of the reserved bandwidth resource while providing probabilistic service guarantees to users.

2. SYSTEM OVERVIEW

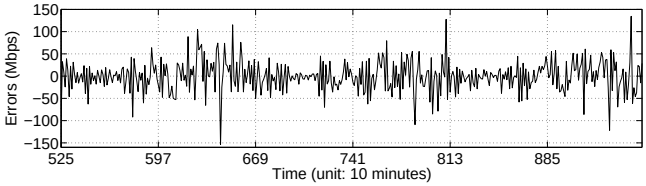
We propose a predictive and adaptive workload management system that works periodically at 10-minute frequency, alternating between phases of *demand estimation* and *workload management*. Suppose there are N video channels and S (possibly distributed) servers.

In the demand estimation phase, we monitor the total number of bytes streamed from servers in each video channel i in each 10-minute period, from which the average bandwidth demand of channel i is calculated. Based on demand history, the expected bandwidth requirement of each channel in the next 10 minutes is predicted. Demand prediction is fundamentally backed up by the fact that the population in a video channel follows diurnal evolution [**?**]. The system not only forecasts the expected demand, but also outputs a *volatility* estimate which represents the degree that the demand may fluctuate around its expectation, as well as the demand *correlation* between different channels.

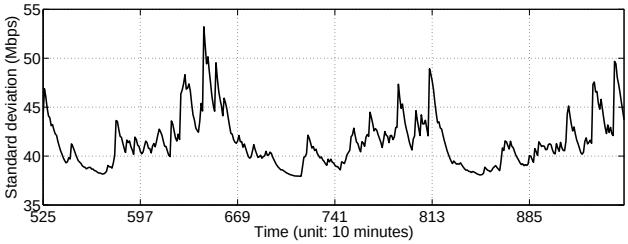
We illustrate demand forecast using real-world traces. Our demand traces come from UUSee [**?**], an operational large-scale VoD service based in China. The dataset contains bandwidth demands in UUSee video channels sampled every 10 minutes during 2008 Summer Olympics. We make 10-minutes-ahead (one-step-ahead) prediction of bandwidth demand $\{S_t\}$ in a typical video channel released at time period $t_0 = 264$ (2008-08-10 10:47:39) in UUSee [**?**, **?**]. The channel has a maximum online population of 2664. The bandwidth consumption series of the first 1.25 days is used as the training data starting from time period 81. The initial 80 time periods are excluded which may not conform to later evolution patterns. We use the seasonal ARIMA model [**?**] to predict the expected bandwidth demand in each time period. The test period is the 3 days following the training period. Fig. 1(a) shows the prediction results, with zero-mean prediction errors plotted in Fig. 1(b). We further use the GARCH model elaborated in [**?**] to predict demand variation around its mean. Fig. 1(c) shows the estimated time-varying demand standard deviation relative to its expectation. It will be clear that the reserved bandwidth for a workload will be its expected demand plus a risk premium depending on the demand variation.



(a) Prediction for the expected demand



(b) Prediction errors



(c) Prediction for the demand standard deviation

Figure 1: 10-minutes-ahead prediction for bandwidth demand S_t in a popular UUSee video channel.

In the workload management phase, the system takes the predicted statistics about future demands as the input and generates a *load direction matrix* $\mathbf{W} = [w_{si}]$, where w_{si} represents the proportion of channel i's workload directed to server s. The output matrix $\mathbf{W}$ essentially corresponds to a joint decision of bandwidth reservation, content placement and load direction, updated every 10 minutes. If a server s has $w_{si} = 0$ for all i, the server is not employed. Thus, the total number of servers used can be determined from $\mathbf{W}$. Moreover, video i is replicated to server s only if $w_{si} > 0$, constituting a content placement decision. Apparently, we have $\sum_s w_{si} = 1$ if the aggregate server capacity is sufficient. In practice, load direction matrix $\mathbf{W}$ can be readily implemented by routing the requests for video channel i to server s with probability w_{si}.

Afterwards, the system proceeds to demand estimation again for the next 10 minutes. It strives to ensure probabilistic service guarantees so that the load imposed on each server does not exceed its bandwidth capacity with a high probability, *e.g.*, 0.98. Under quantitatively confined risks, the system aims to reserve as little bandwidth as possible to enhance resource efficiency. This is achieved by consolidating workloads onto multiple servers based on anti-correlation through mean-variance optimization.

3. WORKLOAD MANAGEMENT

In this section, we focus on using demand statistics and predictions to guide workload management to enhance utilization while hedging server overload risk. Suppose before time t, we have obtained the estimates about demands in the coming $\Delta t = 10$ minutes. Our objective is to decide load direction matrix $\mathbf{W}$ so as to minimize total bandwidth reservation while confining risk of overload at each server.

We first introduce a few useful notations. Recall that N denotes the number of videos in the system. Let random variable D_i denote the bandwidth demand for video i, with estimated expectation $\mu_i = \mathbf{E}[D_i]$ and variance $\sigma_i^2 = \mathbf{Var}[D_i]$, $i = 1, \ldots, N$. Note that the random demands $D_1, \ldots, D_N$ may be highly correlated due to the correlation between video genres, viewer preferences and video release times. Denote ρ_{ij} the correlation coefficient of D_i and D_j, with $\rho_{ii} \equiv 1$. For convenience, let $\boldsymbol{\mu} = [\mu_1, \ldots, \mu_N]^\mathsf{T}$, and $\boldsymbol{\Sigma} = [\sigma_{ij}]$ be the $N \times N$ symmetric covariance matrix with $\sigma_{ii} = \sigma_i^2$ and $\sigma_{ij} = \rho_{ij}\sigma_i\sigma_j$ for $i \neq j$.

Consider S available servers in the system. Denote C_s the upper bound on bandwidth capacity available on server s, $s = 1, \ldots, S$. C_s may be limited by the egress bandwidth capacity of server s. It can also be manually configured to spread the workload across multiple servers and avoid booking bandwidth from a single server.

Recall that the load direction matrix is denoted $\mathbf{W} = [w_{si}]$, $s = 1, \ldots, S$, $i = 1, \ldots, N$, where w_{si}represents the proportion of video i's demand directed to and served by server s, with $\sum_s w_{si} = 1$. We define $\mathbf{w}_s := [w_{s1}, \ldots, w_{sN}]^\mathsf{T}$ as the *workload portfolio* of server s. Given $\mathbf{w}_s$, the aggregate load imposed on server s is a random variable $L_s = \sum_{i=1}^N w_{si}D_i$, with expectation and variance given by

$$\mathbf{E}[L_s] = \mathbf{E}[\textstyle\sum_{i=1}^N w_{si}D_i] = \mu_1 w_{s1} + \ldots + \mu_N w_{sN} = \boldsymbol{\mu}^\mathsf{T}\mathbf{w}_s,$$
$$\mathbf{Var}[L_s] = \mathbf{Var}[\textstyle\sum_{i=1}^N w_{si}D_i] = \sum_{i,j} \rho_{ij}\sigma_i\sigma_j w_{si}w_{sj} = \mathbf{w}_s^\mathsf{T}\boldsymbol{\Sigma}\mathbf{w}_s.$$

3.1 A Single Server's Decision

When a single server s forms its workload portfolio, it aims to serve as many demands as possible, while making sure that enough egress server bandwidth is allocated. For random demands, this goal corresponds to maximizing the expected load $E[L_s]$, while confining $\Pr(L_s > C_s)$ to a small value δ. We thus obtain the following optimization problem:

$$\max \boldsymbol{\mu}^\mathsf{T}\mathbf{w}_s, \tag{1}$$
$$\text{s.t.} \quad \Pr(L_s > C_s) \leq \delta, \tag{2}$$
$$\mathbf{0} \preceq \mathbf{w}_s \preceq \mathbf{1}, \tag{3}$$

where $\mathbf{0} = [0, \ldots, 0]^\mathsf{T}$, $\mathbf{1} = [1, \ldots, 1]^\mathsf{T}$ are N-dimensional vectors.

When L_s is normally distributed (which can be observed from the UUSee traces), constraint (2) is equivalent to

$$\mathbf{E}[L_s] + \theta\sqrt{\mathbf{Var}[L_s]} = \boldsymbol{\mu}^\mathsf{T}\mathbf{w}_s + \theta\sqrt{\mathbf{w}_s^\mathsf{T}\boldsymbol{\Sigma}\mathbf{w}_s} \leq C_s, \tag{4}$$

where $F(\cdot)$ is the CDF of normal distribution $\mathcal{N}(0, 1)$, and $\theta = F^{-1}(1 - \delta)$ is a constant.

To serve the load L_s, $\mathbf{E}[L_s] + \theta\sqrt{\mathbf{Var}[L_s]}$ bandwidth is reserved on server s, where we call $\theta\sqrt{\mathbf{Var}[L_s]}$ the "cushion bandwidth" needed to accommodate load fluctuation. Constraint (4) essentially requires that the total booked resource be bounded by C_s. Note that problem (1) is a second-order conic program and can be solved efficiently using interior point method [**?**].

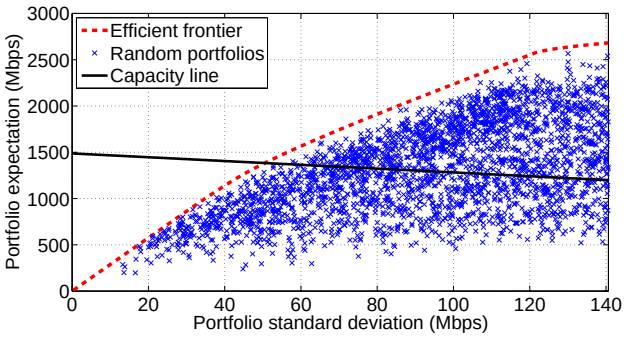


Figure 2: The efficient frontier for portfolios of 5 videos and a single server s on the $\sqrt{\mathbf{Var}[L_s]}$-$\mathbf{E}(L_s)$ plane. Random portfolios are formed by uniformly choosing w_{si} in $(0, 1)$ for $i = 1, \ldots, 5$. The capacity line represents (4) with $C_s = 1500$ Mbps and $\delta = 2\%$.

Alternatively, we can explain problem (1) in terms of the efficient portfolio frequently used in investment theory [**?**]. Among all $\mathbf{w}_s$ that achieves a certain $\mathbf{E}[L_s]$, there is a $\mathbf{w}_s^*$ that leads to the minimum load standard deviation $\sqrt{\mathbf{Var}[L_s]}$. Portfolio $\mathbf{w}_s^*$ is preferred because it achieves the least load variation and thus requires the least cushion bandwidth to be reserved. Therefore, the curve of the minimum $\sqrt{\mathbf{Var}[L_s]}$ as a function of $\mathbf{E}[L_s]$ is called the *efficient frontier*. In a $\sqrt{\mathbf{Var}[L_s]}$-$\mathbf{E}(L_s)$ plane, every possible portfolio is represented by a point lying on or below the efficient frontier.

For example, we consider 5 video channels, with known demand expectation and covariance matrix. The efficient frontier is plotted in Fig. 2, where we also plot random portfolios formed by uniformly choosing w_{si} from $(0, 1)$. In addition, there is a capacity line corresponding to the QoS constraint (4), which passes $(0, C_s)$. The intersection of the capacity line with efficient frontier leads to the optimal solution to problem (1), i.e., the maximum expected load serviceable under a very small overload risk. For example, in Fig. 2, we have $\delta = 2\%$ and $\theta = 2.05$, which means that with probability $F(\theta) = 0.98$, the load on server s is satisfied.

3.2 Multiple Servers

When S servers share the load, we can let the servers optimize their workload portfolios one after another as follows. Initially, let $s = 1$ and solve (1) to obtain $\mathbf{w}_1^*$. We then update D_i to $(1 - \sum_{r=1}^s w_{ri}^*)D_i$ for all i and calculate the new $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ based on the new D_i's. Afterwards, we can update s to $s + 1$, and re-solve (1) to obtain $\mathbf{w}_{s+1}^*$ for the next server. Such a process is repeated until $s = S$.

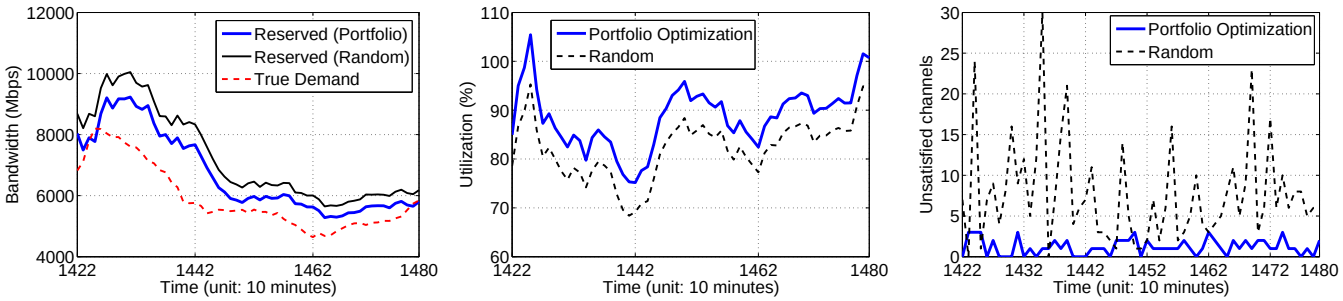
As a result, the bandwidth to be reserved from server s is

$$\boldsymbol{\mu}^\mathsf{T}\mathbf{w}_s^* + \theta\sqrt{\mathbf{w}_s^{*\mathsf{T}}\boldsymbol{\Sigma}\mathbf{w}_s^*}, \tag{5}$$

and the aggregate bandwidth reserved from all servers is

$$S_{\text{booked}} = \textstyle\sum_{s=1}^S (\boldsymbol{\mu}^\mathsf{T}\mathbf{w}_s^* + \theta\sqrt{\mathbf{w}_s^{*\mathsf{T}}\boldsymbol{\Sigma}\mathbf{w}_s^*}), \tag{6}$$

The algorithm enhances the utilization of each server, one after another, by pushing the cushion bandwidth $\theta\sqrt{\mathbf{w}_s^\mathsf{T}\boldsymbol{\Sigma}\mathbf{w}_s}$ on each server to the minimum. As each server s accommodates more expected demand $\boldsymbol{\mu}^\mathsf{T}\mathbf{w}_s$, the necessary aggregate bandwidth reservation is reduced.



(a) Reserved Bandwidth vs. True Demand (b) Utilization of Reserved Bandwidth (c) Channels with Unsatisfied Requests

Figure 3: Workload portfolio optimization vs. random load direction for a typical peak period.

4. PERFORMANCE EVALUATION

We conduct trace-driven simulations to evaluate the performance of workload portfolio optimization, in comparison with *random load direction* described below. Initially, let $s = 1$. We randomly generate $\mathbf{w}_s$ in $(\mathbf{0}, \mathbf{1})$, and rescale it to $\mathbf{w}'_s$ such that QoS constraint (4) is achieved with equality. Update $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ in the same way as in workload portfolio optimization. We then update s to $s+1$, and repeat the above process to obtain $\mathbf{w}'_{s+1}$ for the next server until $s = S$.

In our evaluation, bandwidth reservation is carried out online every $\Delta t = 10$ minutes. Before time t, the system has obtained estimates $\boldsymbol{\mu}_t = [\mu_{1t}, \ldots, \mu_{Nt}]$ and $\boldsymbol{\Sigma}_t = [\sigma_{ijt}]$ for the coming period $[t, t+\Delta t)$ using forecasting methods mentioned in Sec. 2. We then take $\boldsymbol{\mu}_t$ and $\boldsymbol{\Sigma}_t$ as inputs and calculate $\mathbf{W}_t$ under both workload portfolio optimization and random load direction. Based on $\mathbf{W}_t$, the bandwidth to be reserved from each server is calculated. Once completed, the above demand prediction and workload management process is repeated for the next period $[t + \Delta t, t + 2\Delta t)$. To intentionally distribute workloads across multiple servers, we set the available capacity of each server to be 300 Mbps. We set $\theta = 2.05$ to confine the under-provision probability below $\delta = 2\%$ on each server.

We consider time periods 1422—1480, a typical peak usage period, containing 161 concurrent channels (after certain aggregation). Fig. 3 shows a detailed comparison between workload portfolio optimization and random load direction. Fig. 3(a) plots the aggregate reserved bandwidth under both methods as well as the true demand. We see that the aggregate reserved bandwidth under random load direction is larger than that under portfolio optimization. The former also represents significant over-provisioning as compared to the true demand. Fig. 3(b) further plots the utilization of the aggregate reserved bandwidth by true demand under both methods, substantiating the fact that portfolio optimization enhances resource efficiency.

Based on Fig. 3(a) and Fig. 3(b), under-provision does occur if the utilization exceeds 100% or if the aggregate reserved bandwidth is less than the true demand, but not vice versa: even if the aggregate reserved bandwidth suffices, provision shortage can still occur in individual channels. This motivates us to check the number of unsatisfied channels (with provision shortage) at each point of time. We observed that portfolio optimization guarantees better QoS than random load direction, although δ is set to 2% for both methods. The former confines unsatisfied channels to less than 5 throughout the period.

5. CONCLUDING REMARKS

In this position paper, we propose a predictive and dynamic bandwidth management framework for VoD servers, focusing on controlling quality risks and enhancing resource efficiency. The system predicts the expected future demand as well as demand volatility in each video channel, and estimates demand correlations between channels, by monitoring and learning from demand history. Based on demand prediction, the system jointly makes decisions regarding the capacity to be reserved from each server, content placement, as well as load direction across distributed servers.

Borrowing insights from correlation-based hedging techniques in investment theory, our system strives to leverage demand anti-correlation between video channels to consolidate the workload, thus saving the total bandwidth reservation required while diversifying out risks of under-provision or server overload. Specifically, we formulate the workload portfolio management problem as a mean-variance optimization with probabilistic service level guarantees. We evaluate the performance of the proposed algorithm against random workload mixing through simulations driven by a real-world video access dataset from UUSee VoD services.