A Complete Framework for Modelling and Generating Workload Volatility of a VoD System

Shubhabrata Roy, Thomas Begin and Paulo Gonçalves
Inria, ENS Lyon, UCB Lyon 1, UMR 5668, 46 Allée d'Italie, 69007 Lyon, France
Email: {shubhabrata.roy, thomas.begin, paulo.goncalves}@ens-lyon.fr

***Abstract*—We propose a modelling framework, that can be used to reproduce the workload volatility of a Video on Demand (VoD) system. Based on numerical simulations, we evaluate the precision of the estimation procedure we derive to calibrate our parametric model. We also compare its performance to that of other existing models examining the goodness-of-fit of the steady state distribution and of the autocorrelation function of real workload traces. We then give each parameter of the model an interpretation in terms of the workload volatility, that enlightens on some origins of the system dynamics, like the users behaviour.**

***Keywords*—Traffic Model; Calibration; Burstiness; VoD.**

I. Introduction

In recent trends of data-intensive applications with pay-as-you-go execution in a cloud environment, there are new challenges in system management and design to optimize the resource utilization. For instance, some applications exhibit bursty workloads, which lead to highly varying demand in resources. Research to better understand and to faithfully reproduce this demand volatility in simulators calls for pertinent workload models. Past research on traffic modelling has yielded significant results for various types of applications such as Web, P2P or Video streaming. In all these cases, the developed traffic models have served as valuable inputs to assess the efficiency of adapted management techniques. In this work we consider a Video on Demand (VoD) system as a paradigm of applications subject to highly variable demand and we elaborate a complete modelling framework able to reproduce similar bursty workload.

A VoD service delivers video contents to consumers on request. According to Internet usage trends, users are increasingly getting more involved in the VoD and this enthusiasm is likely to grow. According to [1] a popular VoD provider like Netflix alone represents 28% of all and 33% of peak downstream Internet traffic on fixed access links in North America, with further rapid growth expected. IT giants like Apple, Adobe, Akamai and Microsoft are also emerging as competitive VoD providers in this challenging, yet lucrative market. Since VoD has stringent streaming rate requirements, each VoD provider needs to reserve a sufficient amount of server outgoing bandwidth to sustain continuous media delivery (we are not considering IP multicast here). However, resource reservation is very challenging when a video becomes popular very quickly (i.e. buzz) and yields a *flood* of user requests on the VoD servers. To help the providers anticipating these situations, descriptive models are sensible approaches to capture and to get a better insight into the mechanisms that underlie the applications. The goal of a model is then to reproduce, under controlled and reproducible conditions, the behaviour of real systems and to generate workloads that can eventually be used to evaluate the performance of management policies. One important tenet in modelling is to get a parsimonious description (with few parameters that can easily be calibrated from the observation) covering a large diversity in users practices. Moreover, due to the stationarity and uniformity of a model it might be possible to run shorter simulation and still obtain convergent results [2]. In this paper, we elaborate a complete modelling framework that formalises the users behaviour in a VoD system and so, permits to simulate realistic workload regarding burstiness amplitude and dynamics. Our contribution comprises: *(i)* the construction of an epidemic-inspired model adapted to VoD mechanisms; *(ii)* an heuristic procedure to estimate the models parameters from a workload trace; *(iii)* a comparative study emphasising the good match between our model and real VoD workload traces; that we develop in the next three sections, respectively. We discuss related works and draw conclusions in Section V.

This work has been supported by the EU FP7 project SAIL.

II. Proposed Video on Demand (VoD) Model

Following the trails of related works, our model is *inspired* by epidemic models to represent the way information spreads among the viewers (gossip-like phenomenon) in a VoD system. Epidemic spreading models commonly subdivide a population into several compartments: susceptible (noted **S**) to designate the persons who can get infected, and contagious (noted **C**) for the persons who have contracted the disease. This contagious class can further be categorized into two parts: the infected subclass (**I**) corresponding to the persons who are currently suffering from the disease and can spread it, and the recovered class (**R**) for those who got cured and do not spread the disease anymore [3]. In these models $S(t)_{t\geq 0}$, $I(t)_{t\geq 0}$ and $R(t)_{t\geq 0}$ are stochastic processes representing the time evolution of susceptible, infected and recovered populations respectively.

In the case of a VoD system, infected **I** refers to the people who are currently watching the video and can pass the information along. In our setting, $I(t)$ directly represents the current workload which is the current aggregated video requests from the users. Here, we consider the workload as the total number of current viewers, but it can also refer to total bandwidth requested at the moment. The class **R** refers to the past viewers. In contrast to the classical epidemic case our model does not exhibit a threshold phenomenon, i.e. if the initial infected population exceeds a critical threshold (which quantifies the transmission potential of the disease),

978-1-4673-2480-9/13/ $31.00

then the epidemic spreads, else it dies out. There is no such phenomenon in our proposed model which distinguishes it from a classical epidemic model. Another major distinction of our approach stems from introducing a memory effect in our model, assuming that the **R** compartment can still propagate the gossip during a certain random latency period. We deem this assumption necessary from standard social behavior where people keep talking about a video even after watching it. We also consider the number of susceptible viewers to be *infinite*, since the number of subscribers of the popular VoD service providers can go very high. Like standard epidemic models, our model also follows a stochastic process that satisfies the Markov property and takes values in the state space. Then, within a small time interval dt, the probability for a susceptible individual to turn into an active viewer reads: $\mathbb{P}_{S\to C} = (l + (I(t) + R(t))\beta)\mathrm{d}t + o(\mathrm{d}t)$, where $\beta > 0$ is the rate of information dissemination per unit of time and $l > 0$ fixes the ingress rate of spontaneous viewers. At time t, the instantaneous rate of newly active viewers in the system:

$$\lambda(t) = l + (I(t) + R(t))\beta, \tag{1}$$

corresponds to a non-homogeneous (state-dependant) Poisson process which varies linearly with $I(t)$ and $R(t)$. When $\beta \gg l$, the arrival process induced by peer-to-peer contamination dominates the workload increase, whereas it is the spontaneous viewers arrival process that prevails when $l \gg \beta$. This l also restricts the system reaching in the absorbing state.

Regarding the sojourn time in the (**I**) compartment, we assume that the watch time of a video is an exponentially distributed random variable with mean value γ^{-1}, meaning that viewers leave for the (**R**) class at rate γ. As already mentioned, it also deems reasonable to consider that a past viewer will not keep propagating the gossip about a video indefinitely. Instead, they remain active only for a latency random period that we also assume exponentially distributed with mean value μ^{-1}, after which they leave the system (at rate μ). Without loss of generality we assume that watching time (γ^{-1}) of a video is much smaller compared to the memory (μ^{-1}) persistence, leading to $\mu \ll \gamma$.

Another innovation of our model lies in the buzz generating mechanism: we resort to Hidden Markov Models to randomly switch between the nominal (buzz-free) and the buzz regimes. While $\beta = \beta_1$ in the normal state, the propagation rate (gossip) jumps to $\beta = \beta_2 \gg \beta_1$ in buzz regime, triggering thus a sudden and steep increase of the current demand. Transitions between these two hidden and memoryless Markov states occur with rates a_1 and a_2 respectively and characterize the buzz in terms of frequency and duration. In our context we suppose that $a_1 \ll a_2$, i.e. buzz periods are less frequent and shorter in duration than normal periods. Theoretically, we can generalize the model to include many hidden states. But our result shows (see section IV) only two states suffice to reproduce different types of buzz with peaks and troughs at many scales. With these assumptions, and posing ($I(t) = i$, $R(t) = r$) the current state, Fig. 1 shows the state-transition diagram of the model.

A closed-form expression for the steady-state distribution of the workload (i) of this model seems not to be trivial to derive. However, we can express the analytic mean workload

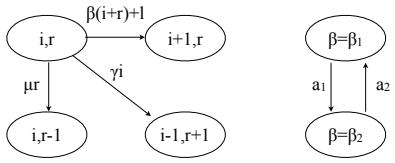

Fig. 1: Markov Chain representing the possible transitions of the number of current (i) and past active (r) viewers.

of the system equaling the incoming and outgoing flow rates in steady regime. For ease, we start with $\beta = \beta_1 = \beta_2$ and generalize the result to $\beta_1 \neq \beta_2$ thereafter. We get: $\mathbb{E}(i) = \mu l/(\mu\gamma - \mu\beta - \gamma\beta)$, which, to be a positive and finite quantity, yields the stability criterion in buzz-free regime:

$$\beta^{-1} > \mu^{-1} + \gamma^{-1}. \tag{2}$$

We now extend these results to the case where the model may exhibit a buzz activity. As β alternates between the hidden states $\beta = \beta_1$ and $\beta = \beta_2$, with respective state probabilities $a_2/(a_1 + a_2)$ and $a_1/(a_1 + a_2)$. Therefore the mean workload in this situation reads:

$$\mathbb{E}(i) = a_2/(a_1 + a_2) \cdot \mathbb{E}_{\beta_1}(i) + a_1/(a_1 + a_2) \cdot \mathbb{E}_{\beta_2}(i), \tag{3}$$

In order to illustrate the flexibility of our workload model and to validate Eq. (3), we generate three synthetic traces corresponding to the different sets of parameters. We reported the result in Table I. Particular realizations of these processes generated over 2^{21} points are displayed in Fig. 2. We choose these three sets of parameters as they lead to three distinct types of workload. The synthetic traces corresponding to cases (b) and (c) reproduce distinct and easily identifiable buzz regimes. However, the buzz shown in case (c) is even more prominent than that occurs in case (b). The parameter set of case (a) leads to a workload variation which is different from case (b) and (c) and the buzzes are not easily identifiable here. Nonetheless, for all 3 configurations, the empirical means estimated from the 2^{21} samples of the traces are in good agreement with the expected values of Eq. (3).

III. Model Calibration

In this section we address the identifiability of our model and design a calibration algorithm to fit workload data (stationary trace). We start constructing empirical estimators for each parameter of the model and then numerically evaluate their performance on synthetic traces.

A. Parameters estimation

Considering a standard epidemic process X with propagation rate θ, the maximum likelihood estimate $\widehat{\theta}_{\mathrm{MLE}}$ is derived

TABLE I: Parameters value used to generate the traces plotted in Fig. 2. The last two rows correspond to the theoretical mean workload of Eq. (3) and to the sample mean value estimated from the traces.

	(a)	(b)	(c)
β_1	4.762×10^{-4}	3.225×10^{-5}	2.439×10^{-5}
β_2	0.0032	0.0032	0.0032
γ	0.0111	0.0020	0.0011
μ	5×10^{-4}	3.289×10^{-5}	2.5×10^{-5}
l	10^{-4}	10^{-4}	10^{-4}
a_1	10^{-7}	10^{-7}	10^{-7}
a_2	0.0667	0.0667	0.0667
$\mathbb{E}(i)$	1.92	16.41	44.93
Emp. mean $\langle i \rangle$	1.74	16.72	45.23

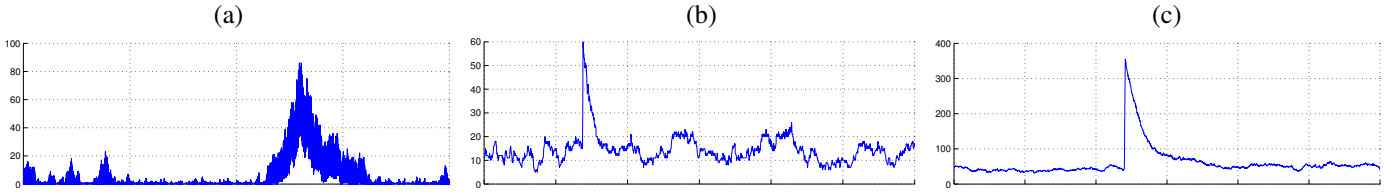

Fig. 2: Illustration of our model ability at generating different dynamics of workload $I(t)$. See Table I for the parameter values corresponding to each of these three cases. The $X-$axis corresponds to time (in hours unit) while the $Y-$axis indicates the number of active viewers.

in [3], [4] and reads:

$$\widehat{\theta}_{\text{MLE}} = n \cdot (\int_0^T X(t)\,\mathrm{d}t)^{-1}, \tag{4}$$

where n is the number of contaminations (i.e. number of increments of X) occurring within the time interval T.

Very often, the maximum likelihood approach yields optimal results (in terms of estimated variance and/or bias) but it is not always possible to get a closed-form expression for the estimated parameters. This can either be due to the likelihood function that is impossible to derive analytically, or to missing data that preclude straightforward application of the maximum likelihood principle. Nonetheless, solutions, such as the Expectation-Maximization (EM) or the Monte Carlo Markov Chain (MCMC) algorithms exist, which in some cases can approximate maximum likelihood estimators.

Returning to our model depicted in Fig. 1, each parameter needs to be empirically estimated, assuming that the instantaneous workload time series is the only available observation.

Watching parameter γ. As γ is the departure rate of users that leave the infected state after they finished watching a video, it can directly be inferred from the number n of decrements of the observable process $I(t)$. Therefore, the MLE of Eq. (4) straightforwardly applies and leads to:

$$\widehat{\gamma}_{\text{MLE}} = n \cdot (\int_0^T I(t)\,\mathrm{d}t)^{-1}. \tag{5}$$

Memory parameter μ. This rate at which past viewers leave the recovery compartment and stop propagating the virus (gossip), relates to the decrement density of the non-observed process $R(t)$. It is thus impossible to simply apply the MLE of Eq. (4) unless we first construct a substitute $\widehat{R}(t)$ to the missing data from the observable data set $I(t)$. Let us recall that in our model, all current viewers turn and remain contagious for a mean period of time $\gamma^{-1} + \mu^{-1}$. Then, in first approximation, we can consider that $R(t)$ derives from the finite memory cumulative process:

$$\widehat{R}(t) = \int_{t-(\gamma^{-1}+\mu^{-1})}^{t} I(u)\,\mathrm{d}u, \tag{6}$$

which itself, depends on the parameter to be estimated μ. We propose an estimation procedure based on the inherent exponential property of the model. From the Poisson assumption, the inter-arrival time $\mathbf{w}$ between the consecutive arrivals of two new viewers is an exponentially distributed random variable such that $\mathbb{E}(\mathbf{w}|\, I(t) + R(t) = x) = (\beta x + l)^{-1}$. It means that, for x fixed, the normalized random variable $\widetilde{\mathbf{w}} = \mathbf{w}/\mathbb{E}(\mathbf{w}|x)$ is exponentially distributed with unitary parameter and becomes independent of x. Ideally then, for each value of $R(t)+I(t) = x$, all the sub-series $\mathrm{w}_x = \{w_n \;:\; R(t_n) + I(t_n) = x\}$, after normalization by their own empirical mean, yield independent and identically distributed realizations of a unitary exponential random variable. In practice though, as $R(t)$ is not observable,

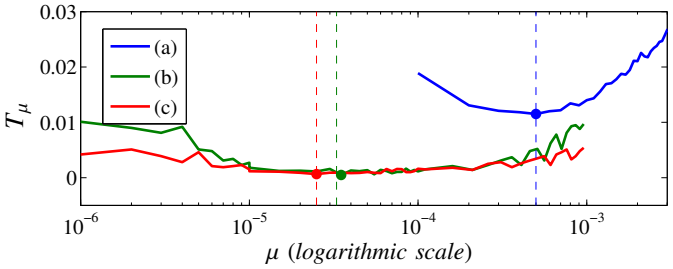

Fig. 3: Evolution of the exponential test statistics (Eq. 7) applied to the traces of Fig. 2. Dotted vertical lines locate the actual value of μ for each case; dot markers on each curve indicate the estimated value $\widehat{\mu}$ corresponding to the minimum point of the statistical test T_μ.

only if $\widehat{R}(t)$ is accurately estimated, should this unitary exponential i.i.d. assumption hold true. From there, we propose the following algorithm: for different values of μ spanning a *reasonable* interval, we use $\widehat{R}_\mu(t)$ estimated from Eq. (6) to build the normalized series $\widetilde{\mathrm{w}}_\mu$. A statistical test applied to each $\widetilde{\mathrm{w}}_\mu$ allows for assessing the exponential i.i.d. hypothesis and then to select the value of μ that yield the best score.

More concretely, we apply to $\widetilde{\mathrm{w}}_\mu = (\widetilde{w}_n)_{n=1,\dots,N}$ the statistical exponentially test derived in [5]: Form the *normalized spacings* $\mathrm{v}_\mu = (v_{(n)} = (N-n+1)(\widetilde{w}_{(n)} - \widetilde{w}_{(n-1)}))_{n=1,\dots,N}$ where $(\widetilde{w}_{(n)})_{n=1,\dots,N}$ stands for $\widetilde{\mathrm{w}}_\mu$ rearranged in ascending order. Let F and G denote the cumulative distribution functions of $\widetilde{\mathrm{w}}_\mu$ and v_μ respectively, and compute the classical Kolmogorov-Smirnov distance:

$$T_\mu = \tfrac{1}{\sqrt{N}} \sup_{1\le k\le N} |F(k) - G(k)|. \tag{7}$$

As F and G are identical for an exponentially i.i.d. random series, we then expect T_μ to reach its minimum for the value of μ that gives the best estimate $\widehat{R}_\mu(t)$ of $R(t)$:

$$\widehat{\mu} = \mathrm{argmin}_\mu T_\mu \text{ and } \widehat{R} = \widehat{R}_{\widehat{\mu}}. \tag{8}$$

Plots of Fig. 3 show the evolution of the Kolmogorov-Smirnov distance corresponding to the traces displayed in Fig. 2. In the 3 cases, T_μ clearly attains its minimum bound for $\widehat{\mu}$ close to to the actual value. The corresponding estimated processes $\widehat{R}(t)$ derived from Eq. (8) match fairly well the real evolution of the (**R**) class in our model (see Fig. 4).

Propagation parameters β and l. According to our model, the arrival rate $\lambda(t)$ of new viewers is given by Eq. (1). It linearly depends on the current number of active and past viewers. So, from the observation $I(t)$ and the reconstructed process $\widehat{R}(t)$ of Eq. (8), we could formally apply the maximum likelihood Eq. (4) to estimate β. In practice however, we have to bear in mind that: *(i)* the arrival process of rate $\lambda(t)$ comprises a spontaneous viewers ingress that is governed by parameter l and which is independent of the current state of the system; *(ii)* depending on the current hidden state of the model (buzz-

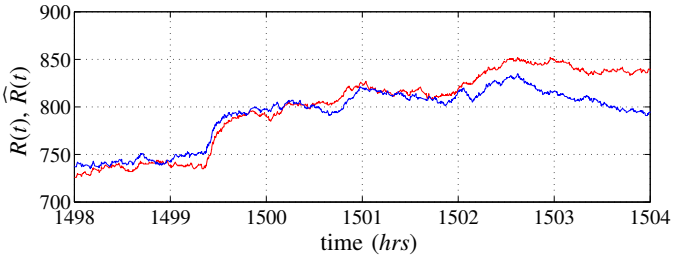

Fig. 4: Evolution of the number of active past viewers. Comparison of the actual (non observable) process $R(t)$ (blue curve) with the estimated process $\widehat{R}(t)$ (red curve) derived from Eq. (6).

free *versus* buzz state), it is alternately $\beta = \beta_1$ and $\beta = \beta_2$ that fix the propagation rate in Eq. (1). We designed an estimation procedure based on a weighted linear regression, that simultaneously addresses these two issues. We decompose our rationale in two steps: First, let us consider the buzz-free state only and $\beta = \beta_1$. As discussed in the estimation of μ the inter-arrival time $\mathbf{w}$ between the consecutive arrivals of two new viewers is an exponentially distributed random variable such that $\mathbb{E}(\mathbf{w}|\, I(t) + R(t) = x) = (\beta\, x + l)^{-1}$. Concretely then, for different values of the sum $I(t) + \widehat{R}(t)$, we calculate the conditional empirical mean: $\Omega(x) = \frac{1}{|\mathcal{I}(x)|} \sum_{t_n \in \mathcal{I}(x)} w_n \;:\; \mathcal{I}(x) = \{t_n : I(t_n) + \widehat{R}(t_n) = x\}$. The linear regression of $(\Omega(x))^{-1}$ against x yields at one go, both parameters estimation $\widehat{\beta}$ (slope) and $\widehat{l}$ (intercept).

Let us now return to the general form of our model with alternation of buzz and buzz-free periods. In the buzz-free case, $\beta = \beta_1$ corresponds to a normal workload activity, meaning that the sum $I(t) + \widehat{R}(t)$ takes on rather moderate values. Conversely, when the system undergoes a buzz, $\beta = \beta_2$ and the population $I(t) + \widehat{R}(t)$ suddenly increases to reach significantly larger values. Yet, in both cases, the quantity Ω^{-1} remains linear with x but with two different regimes (slopes) depending on the amplitude of $I(t) + \widehat{R}(t) = x$. As a result, it is possible to reduce the bias that β_2 causes on the estimation of β_1, using a weighted linear regression of Ω^{-1} *vs* x where the weights $p(x)$ are proportional to the cardinal of the indicator sets $\mathcal{I}(x)$. Indeed, $|\mathcal{I}(x)|$ should be smaller for larger values of x because buzz episodes are expected to be less frequent than nominal activity periods.

Formally, we can apply the exact same procedure to estimate β_2, but considering opposite weights to favor the large values of x's. However, due to the large fluctuations of $(\Omega(x))^{-1}$ in the corresponding region, the slope $\widehat{\beta_2}$ is subject to a very poor estimation variance. Instead, we propose to apply the ML estimator described in Eq. (4) on the restriction of $I(t)$ to the buzz periods only. Strictly speaking, we should consider $\widehat{R}(t)$ as well, but since a buzz event normally occurs on very small interval of time, we assume that $\widehat{R}(t)$ (resp. $R(t)$) remains constant in the meanwhile (flash crowd viewers will enter in R compartment only after the visualization time). In practice, to automatically identify the buzz periods, we threshold $I(t)$ and consider only the persistent increasing parts that remain above the threshold.

Transition rates a_1 and a_2. At time t, the inter-arrival time $\mathbf{w}$ separating to new incomers is a random variable drawn from an exponential law of parameter $\lambda = \beta(i + r) + l$, where β is either equal to β_1 or to β_2. We denote $f_1(\mathbf{w})$ and $f_2(\mathbf{w})$ the corresponding densities built upon the reconstructed process $\widehat{R}(t)$ and the estimated parameters $(\widehat{\beta_1}, \widehat{l})$ and $(\widehat{\beta_2}, \widehat{l})$ respectively. For a given inter-arrival time $\mathbf{w} = w_n$ observed at time t_n, we form the likelihood ratio $f_2(w_n)/f_1(w_n)$ to determine whether the system is in buzz or in buzz-free state. Moreover, in order to avoid non-significant state transitions we resort to a restoration method inspired by the Viterbi algorithm [6]. Once we have identified the hidden states of the process, we estimate the transitions rates $\widehat{a_1}$ and $\widehat{a_2}$ from the average times spent in each state.

B. Numerical Validation

To evaluate the statistical performance of our estimation procedure, we resort to numerical experiments to empirically get the first and the second order moments of each parameter estimator. Owing to the versatility of our model, we must ensure that the proposed calibration algorithm performs well for a variety of workload dynamics. To this end, we systematically reproduce the experiments considering the 3 sets of parameters reported in Table I. For each set, we generate 10 independent realizations of processes similar to the ones depicted in Fig. 2, and use these to derive descriptive statistics. The box plots of Fig. 5 indicate for each estimated parameter (centered and normalized by the corresponding actual value) the sample median (red line), the inter-quartile range (blue box height) along with the extreme samples (whiskers) obtained from time series of length 2^{21} points. As expected (from maximum likelihood), estimation of γ shows to be the most accurate, both in terms of bias and variance. More surprisingly though, although the estimation $\widehat{\beta_1}$ derives from a heuristic procedure that itself depends on the raw approximation $\widehat{R}(t)$ of Eq. (6), the resulting performance is remarkably good: bias is always negligible (less than 5% in the worst case (c)) and the variance always confines to 10% interval. Notice also that the estimation of β_1 goes from a slight underestimation in case (a) to a slight overestimation in case (c), as the buzz effect, i.e. the value of β_2, grows from traces (a) to (c). Compared to $\widehat{\beta_1}$, the estimation of β_2 behaves more poorly and proves to be the hardest parameter to estimate. But we have to keep in mind that this latter is only based on buzz periods which represent only a small fraction of the entire time series. Regarding the parameter μ, its estimation remains within a 20% inter-quartile range but all cases show a systematic bias (median hits the lower or upper quartile bound). Remind that the procedure, described by Eq. (8) to determine $\widehat{\mu}$ selects within some discretized interval, the value of μ that yields the best T_μ score. It is then very likely that the true value does not coincide with any sampled point of the interval and therefore, the procedure picks the closest one that systematically lies beneath or above. Finally, estimation of the transition parameters a_1 and a_2 between the two hidden states relies on all other parameters estimation, cumulating all relative inaccuracies. Nonetheless and despite a systematic underestimating trend, precision remains within a very acceptable confidence interval.

Convergence rate of the empirical estimators is another important feature that binds the estimate precision to the

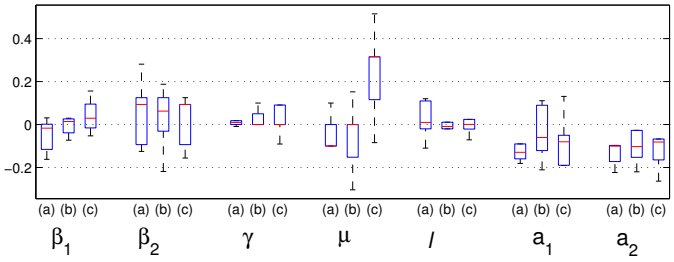

Fig. 5: Relative precision of estimation of the model parameters. Cases (a), (b) and (c) correspond to the configurations reported in Table I. Statistics are computed over 10 independent realizations of time series of length 2^{21} points.

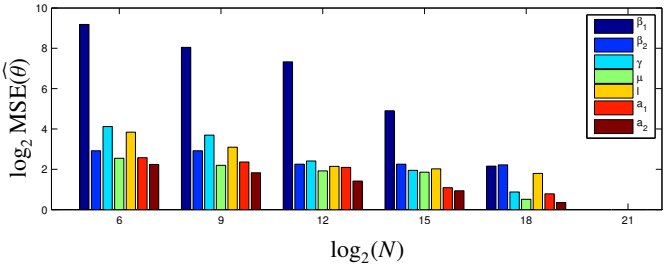

Fig. 6: Evolution of the Mean Square Error *versus* the data length N in a *log-log* plot. For the sake of conciseness, we only show here the results corresponding to the case (b) of Table I.

amount of available data. Using the same data set, the bar plots of Fig. 6 depicts the evolution of the mean square error $\mathrm{MSE}(\widehat{\theta}) = \mathbb{E}\{(\widehat{\theta} - \theta)^2\}$ – where generic θ stands for any parameter of the model – with the length N of the observable time series. As our purpose is to stress the rate of convergence of these quantities towards zero, to ease the comparison, we normalize the MSE of each parameter by its particular value at maximum data length (i.e 2^{21} points here). Then, the estimator rate of convergence α_θ corresponds to the decaying slope of the MSE with respect to N in a *log-log* plot, i.e. $\mathrm{MSE}(\widehat{\theta}) \sim O(N^{-\alpha_\theta})$. For the different parameters of our model we obtain convergence rates that lie between $\alpha_{\beta_1} = 0.5$ and $\alpha_{a_2} = 0.2$, leading each time to sub-optimal convergence ($\alpha_\theta < 1$).

IV. Model Fitting to Real Workload Data

To assess the adequacy of our model at reproducing real workload traces, we apply the calibration procedure described in Section III on two VoD traces, recorded in January 2011 by the Greek Research and Technolohy Network (GRNET) [7]. We denote them as Trace I (∼ 200 hours long) and Trace II (∼ 150 hours long) and plotted in Fig 7-(a) and -(b), respectively. For both cases, we check that the two sets of estimated parameters reported in Table II verify the stability condition of Eq. (2) and we use the so calibrated models to generate two corresponding realisations of synthetic workloads (plots (c)-(d) of Fig. 7).

Comparing the means and the standard deviations of both real and synthetic traces (Table III), it is clear that our model successfully reproduces the average number of active viewers but also its variability along time. The observed difference (about 10% for the mean values) is not as striking as it was with the synthetic traces of Section II. But we must bear in

TABLE II: Estimated Parameters of our VoD model.

	$\widehat{\beta_1}$	$\widehat{\beta_2}$	$\widehat{\gamma}$	$\widehat{\mu}$	$\widehat{l}$	$\widehat{a_1}$	$\widehat{a_2}$
Trace I	$1.3.10^{-3}$	$8.4.10^{-3}$	$3.9.10^{-3}$	$2.8.10^{-3}$	$3.2.10^{-3}$	$3.1.10^{-4}$	$2.2.10^{-2}$
Trace II	$4.9.10^{-3}$	$1.8.10^{-2}$	$1.2.10^{-2}$	$9.5.10^{-3}$	$4.8.10^{-4}$	$1.3.10^{-5}$	$4.1.10^{-2}$

TABLE III: Mean and standard deviation of real traces and models.

		Real	Proposed Model	Simple Markov	$MMPP/M/1$
Trace I	Mean	4.99	5.59	12.68	6.45
	Std. Dev	18.26	17.87	17.15	20.02
Trace II	Mean	0.71	0.62	1.23	0.94
	Std. Dev	16.82	15.99	15.85	17.95

mind that first, *ab initio* nothing guarantees that the underlying system matches our model dynamics and, second, Traces I and II can possibly encompass short scale non-stationary periods (e.g. day *versus* night activity) which are not accounted for in our model.

Nonetheless, for the sake of a fair analysis, we must compare the performance of our approach with that of simpler, yet sensible models and with that of more elaborated models that were proposed in the literature for similar purposes. Then, we start with a simple Markov model where the transition rates derive from all possible changes of states observed in real time series. Calibrated on Traces I and II, this model produces synthetic evolutions of active viewers, represented in Fig. 7(e)-(f), whose mean can significantly differ from real values (see Table III). However the discrepancy is not that pronounced for the standard deviations (relative error remains below 10%), which tends to prove that a naive model like a Markov chain succeeds to catch the inherent variability of a VoD workload process!

Let us now consider the more refine $MMPP/M/1$ queue model proposed in [8]. This queueing system assumes an arrival process that alternates between two Poisson processes according to a two hidden state Markov chain, an exponentially distributed service time and a single server to serve the viewers. In the author's own words, this Modulated Markov Poisson Process is particularly adapted for modelling correlated arrival streams and bursty workload behaviour. As previously then, we calibrate this model with Traces I and II and we plot in Fig. 7(g)-(h) the realisations of the corresponding two synthesised workloads. Comparing the means and the standard deviations between the real and the modelled traces, the fitting performance of the $MMPP/M/1$ model are fairly comparable to that of our model (see values in Table III)...

Beyond its mean and standard deviation, the steady state distribution of a (stationary) stochastic process is a more complete indicator of the process volatility. In particular, the way it decreases towards zero defines the frequency of large values and therefore directly reflects the burstiness of the process. Top and bottom plots of Fig. 8 represent the estimated steady state distributions corresponding to the real workloads of Traces I and II, respectively and superimposed, the ones for the three different models we considered. Despite having comparable means and variances (Table III), these curve show that not all the synthetic traces do reproduce accurately the statistical distribution of the number of active viewers. In particular, it is clear from the plots that the occurrence of large amplitudes are overvalued by the simple Markov model and also by the $MMPP/M/1$ queue. In contrast, the good fit of

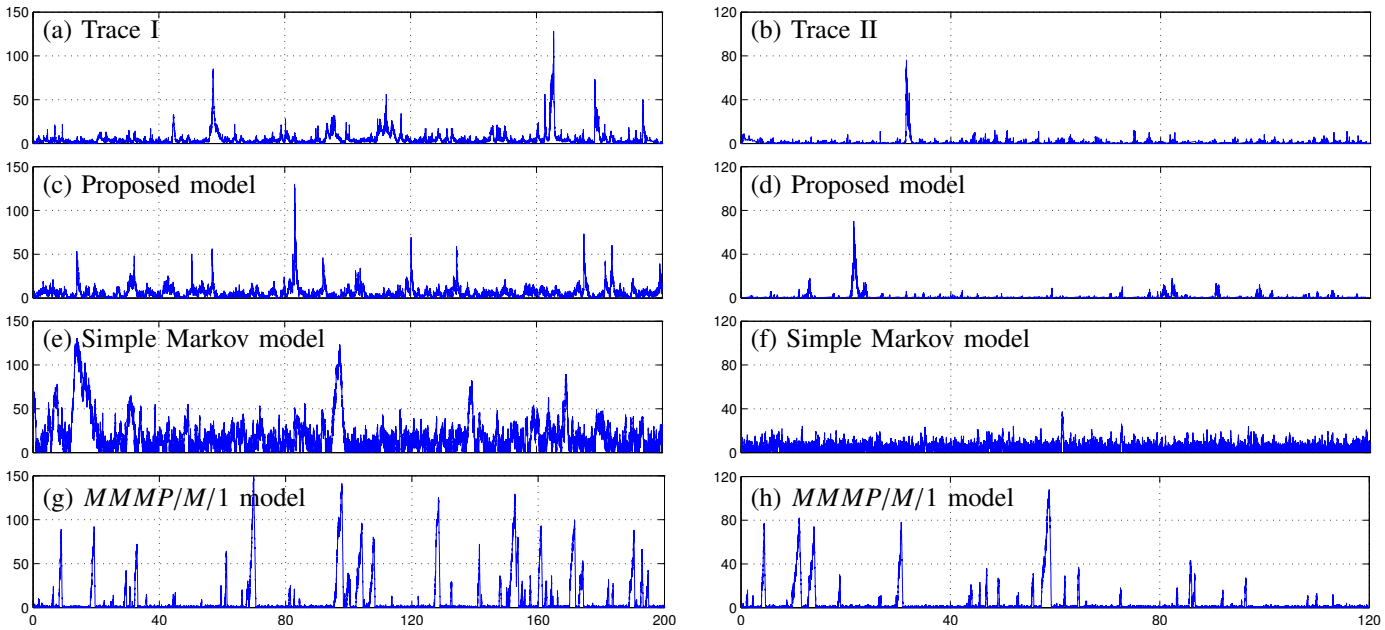

Fig. 7: Modelled workload for Trace I (Left column) and Trace II (Right column). First row corresponds to the real traces; second row to the synthesised traces from our model; third row to synthesised traces from simple Markov model; fourth row to the synthesised traces from MMPP/M/1 model. Horizontal axes represent time (in hours) and vertical axes represent workload (number of active viewers).

our model proves its capacity to reproduce the occurrence and the amplitude range of buzz events (i.e. bursts in the evolution of active viewers).

Another very important feature that characterises the volatility of a process is the local regularity of its path. In particular, the rapidity of the amplitude variations at small scales fixes the dynamics of the bursts, and can subtly be formalised via the auto-correlation function of the process. This latter measures the statistical dependency $R_I(\tau) = \mathbb{E}\{I(t)\, I^*(t+\tau)\}$ between two samples of a (stationary) process I, distant of a time lag τ: the larger $R_I(\tau)$, the smoother the path of I at scale τ. So, for all the trajectories of Fig. 7, we estimated their auto-correlation functions that we plotted in Fig. 9. It is striking then, how our model is able to reproduce the long-term correlative structure of the real traces, whereas both simple Markov and $MMPP/M/1$ models fail at imposing a statistical continuity beyond a 30 minutes time scale for Trace I, and only 3 minutes for Trace II! Actually, this time coherence is also very visible in the synthetic traces of our model (Fig. 7(c),(d)) that look much smoother compared to the very erratic ones of the other models (Fig. 7(e)-(h)).

Let us stress that this reproduced dynamics is a direct consequence of the memory effect (controlled by the parameter μ) we injected in our model. However, we did not intend with this mechanism, to originate a Long Range Dependence (LRD) property (in the strict sense of a power law decay of the autocorrelation function), as we did not observe such behaviour in real data. This explains also, why we did not report our experiments based on a Lévy process model. Despite its ability at matching highly varying processes, this model is specifically known for exhibiting LRD that is incompatible with the sought autocorrelation decay.

Finally, owing to its constructive approach, each parameter of our model is meaningful with regards to the system operation. For instance, let us compare the values of parameters $\widehat{\beta_1}$ and $\widehat{l}$ estimated from Trace I and II. In the first case, the arrival of new viewers is dominated by spontaneous incomers and is not so much due to information propagation through gossip (except in the buzz periods), whereas in the second trace, the peer-to-peer diffusion component overtakes the spontaneous attraction of the server. At the same time, the memory index $\widehat{\mu}$ tripled, meaning that the mean duration of contagion shrank by a factor of 3. This parameter could then be used as an indicator of the content interest delivered by the server, through its lifetime in users mind. A VoD service providers can exploit these information for better provisioning the system [9].

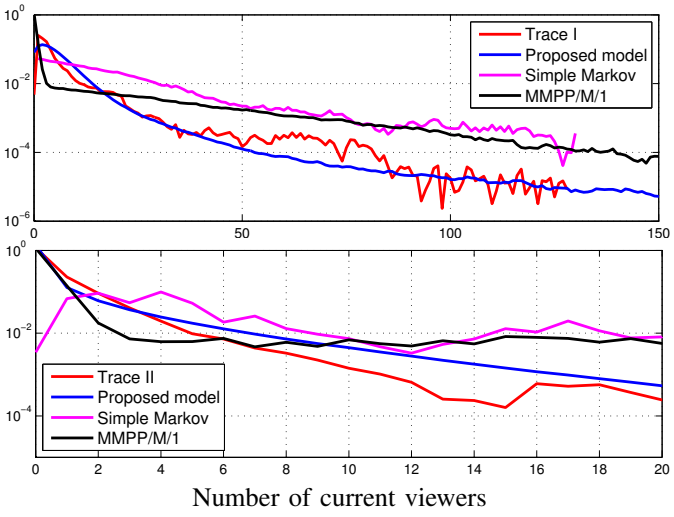

Fig. 8: Steady-state distribution of the real and generated traces from the proposed, simple Markov and $MMPP/M/1$ models. Top plots corresponds to Trace I and bottom to Trace II.

V. Related Works and Conclusion

A survey of the history of VoD modelling shows several changes of paradigms and platforms. An early work in this

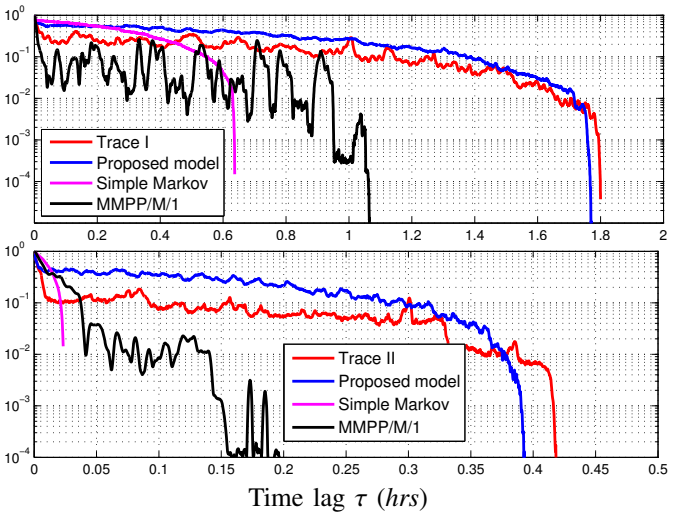

Fig. 9: Empirical autocorrelation function of the real and generated traces from the proposed, simple Markov and $MMPP/M/1$ models. Top plots corresponds to Trace I and bottom to Trace II.

domain include [10]. But, we focus on the bursty workload generated by a VoD system, which has been an active area of research with different approaches. In an user based approach authors of [12], [13] and [14] develop user activity models to reproduce the workload generated by an user. In a different approach researchers [15] [16] aim to model the aggregated workload generated by multiple users. In this vein we discuss some basic as well as advanced models which address workload volatility of a VoD or similar systems in the next paragraph. Authors of [17] proposed a maximum likelihood method for fitting a Markov Arrival Process (MAP), a generalization of the Poisson process by having non-exponentially distributed (yet phase type) inter-arrival times, to the web traffic measurements. This approach is useful to describe the time-varying characteristics of workloads and seems to achieve reasonable accuracy in models to fit web server traces in terms of inter-arrival times and tail heaviness. However, the authors do not aim to model bursty workloads in this work. With a focus on buzz arrival modelling, the authors of [15] and [16] proposed a two-state $MMPP$ (a special case of MAP) based approach and a parameter estimation procedure using the index of dispersion. But as we saw in section IV the $MMPP$ model seems to include only very short memory and may not be suitable for our purpose. Moreover, the obtained model parameters from both MAP and $MMPP$ are not comprehensive to draw inference about the system dynamics. A parsimonious model like Lévy is a tempting approach since it can provide a long-term correlation of the system. Thanks to its inherent "α-stable" process, this process is also suitable to model system volatilities. But it develops a long range memory which does not seem to match the dynamic feature of our real traces. In a distinct approach, server workloads have been thoroughly studied in many works, such as [18], [11], [19], [20] or [21]. These works, however, provide a statistical analysis of server workloads in context of usage pattern, caching, pre-fetching, or content distribution and do not focus primarily on workload modelling. Other popular workload generators include [22], [23], [24] or [25]. They are mostly used to evaluate computer systems or Web sites. But it seems that reproducing satisfactory burstiness in workload is the major deficiency of most proposed models.

Our approach demarcates itself from the above described works mainly in its constructiveness and ability to generate bursty traffic with correct dynamics. We propose a comprehensive model, capable to reproduce workload volatility with "avalanche" type of burstiness. Moreover, this model is versatile and can be adapted to other applications as well. Finally, owing to the constructive nature of our model, the estimated values of the parameters provide valuable insight on the application that is difficult to infer readily from the raw traces. The captured information may answer questions of practical interest to cloud oriented VoD service providers which we intend to explore as our potential future work.

References

[1] Sandvine, "http://www.sandvine.com/news/pr_detail.asp?ID=312/," .

[2] L. Eeckhout, K. De Bosschere, and H. Neefs, "Performance analysis through synthetic trace generation," in *IEEE ISPASS*, 2000, pp. 1–6.

[3] A. Barrat, M. Barthelemy, and A. Vespignani, *Dynamical Processes on Complex Networks*, Cambridge University Press, 2008.

[4] H. Andersson and T. Britton, *Stochastic Epidemic Models and Their Statistical Analysis*, vol. 151, Springer, Lecture Notes in Statistics, 2000.

[5] S.R. Jammalamadaka and E. Taufer, "Testing exponentiality by comparing the empirical distribution function of the normalized spacings with that of the original data," *J. Nonparametric Statistics*, vol. 15, 2003.

[6] J. Kleinberg, "Bursty and hierarchical structure in streams," in *ACM SIGKDD*, 2002.

[7] GRNET, "Video traces obtained from grnet," Accessed July 2011, http://vod.grnet.gr/.

[8] J. Revzina, "Possibilities of MMPP processes for bursty traffic analysis," in *RELSTAT*, 2010.

[9] P. Gonçalves, S. Roy, T. Begin, and P. Loiseau, "Dynamic resource management in clouds: A probabilistic approach.," *IEICE*, 2012.

[10] G. Bianchi and R. Melen, "The role of local storage in supporting video retrieval services on atm networks," *IEEE Networking*, 1997.

[11] M. Naldi, "A mixture model for the connection holding times in the video-on-demand service," *Performance Evaluation*, 2002.

[12] V.O.K. Li, W. Lao, X. Qiu, and E. W. M. Wong, "Performance model of interactive video-on-demand systems," *IEEE JSAC*, vol. 14, 1996.

[13] D. Melendi, R. Garcia, X. G. Paneda, and V. Garcia, "Multivariate distributions for workload generation in video on demand systems," *IEEE Comm. Letters*, vol. 13, 2009.

[14] S. Kanrar, "Analysis and implementation of the large scale video-on-demand system," *IJAIS*, vol. 2, no. 2, 2012.

[15] D. Perez-Palacin, J. Merseguer, and R. Mirandola, "Analysis of bursty workload-aware self-adaptive systems," in *ICPE*, 2012.

[16] R. Gusella, "Characterizing the variability of arrival processes with indexes of dispersion," *IEEE JSAC*, vol. 9, no. 2, 1991.

[17] S. Pacheco-Sanchez, G. Casale, B. Scotney, S. McClean, G. Parr, and S. Dawson, "Markovian workload characterization for QoS prediction in the cloud," in *IEEE Cloud*, 2011.

[18] M. Arlitt and T. Jin, "Workload characterization of the 1998 world cup web site," Itechnical report hpl-1999-35r1, HP Labs, 1999.

[19] X. Chen and X. Zhang, "A popularity-based prediction model for web prefetching," *IEEE Computer*, 2003.

[20] E. Caron, F. Desprez, and A. Muresan, "Pattern matching based forecast of non-periodic repetitive behavior for cloud clients," *J. of Grid Comp.*, 2011.

[21] R. Garcia, X. Paneda, V. Garcia, D. Melendi, and M. Vilas, "Statistical characterization of a real video on demand service: User behaviour and streaming-media workload analysis," *Simul Model Pract Th*, 2007.

[22] Faban, "Faban project web site," http://faban.sunsource.net/.

[23] Httperf, "Httperf project web site," http://code.google.com/p/httperf/.

[24] Jmeter, "Jmeter project web site," http://jakarta.apache.org/jmeter/.

[25] P. Barford and M. Crovella, "Generating representative web workloads for network and server performance evaluation," in *SIGMETRICS*, 1998.