Open access • Proceedings Article • DOI:10.1109/PACKET.2007.4397033

Comparison of simulcast and scalable video coding in terms of the required capacity in an IPTV network — Source link

Zlatka Avramova, Danny De Vleeschauwer, Kathleen Spaey, Sabine Wittevrongel ...+2 more authors

Institutions: Ghent University, Alcatel-Lucent, University of Antwerp

Published on: 06 Dec 2007

Topics: Scalable Video Coding, Video quality, IPTV and Multicast

Comparison of simulcast and scalable video coding in terms of the required capacity in an IPTV network

Comparison of Simulcast and Scalable Video Coding in Terms of the Required Capacity in an IPTV Network

Z. Avramova*, D. De Vleeschauwer #,*, K. Spaey†, S. Wittevrongel*, H. Bruneel*, C. Blondia†
* SMACS Research Group, TELIN Department, Ghent University, Sint-Pietersnieuwstraat 41, B-9000 Ghent, Belgium
† PATS Research Group, IBBT-University of Antwerp, Middelheimlaan 1, B-2020 Antwerp, Belgium
# Alcatel-Lucent Bell, Copernicuslaan 50, B-2018 Antwerp, Belgium

Abstract—**The diversity of multimedia-enabled devices supporting streamed multimedia is ever growing. Multicast delivery of TV channels in IP networks to a heterogeneous set of clients can be organised in many different ways, which brings up the discussion which one is optimal. Scalable video streaming has been believed to be more efficient in terms of network capacity utilisation than simulcast video delivery because one flow can serve all terminals, while with simulcast all resolutions are offered in parallel. At the same time, it is also largely recognised that in order to provide the same video quality compared to non-layered video coding, scalable video coding (SVC) incurs a bit rate penalty.**

In this paper we compare simulcast and SVC in terms of their required capacity in an IPTV network scenario where a bouquet of TV channels is offered to the subscribers. We develop methods to calculate and approximate the capacity demand for two different subscriber behaviour models. These methods are then used to explore the influence of various parameters: the SVC bit rate penalty, the number of offered channels, the channel popularity and the number of subscribers. The main contribution of this paper is that we derive an analytical formula to calculate the SVC limit bit rate penalty beyond which SVC is less efficient than simulcast. In the realistic IPTV examples considered here, the limit is found to lie between 16% and 20%, while the reported values for this coding penalty range from 10% up to 30% for current H.264 SVC codecs, indicating that SVC in IPTV is not always more efficient than simulcast.

I. Introduction

A. Background

IPTV has taken off the last years and is becoming more and more widely deployed. Streaming multimedia and broadcasting TV and radio channels through IP-core networks and mobile networks are already in place worldwide. TV and video (multimedia) broadcasting will become ubiquitously digital if not all-IP-transported in the near future. The shift to all-digital networks brings numerous benefits, such as several times more efficient utilisation of bandwidth in cable distribution plants, convergence of services, more flexible capacity allocation and utilisation in the presence of a feedback channel, etc.

This work was carried out partly within the framework of the IWT (Flemish Institute for the promotion of Scientific and Technological Research in the Industry) project Q-MATCH.

In order to meet the heterogeneity requirements of converged networks and/or the terminal diversity of the users, video must be adaptively delivered. Such scenarios arise in heterogeneous IP networks, surveillance systems, mobile streaming media networks, wireless video LANs, multi-party video conferencing systems, etc. When network broadcast operators deliver TV channels to heterogeneous clients (heterogeneous by their devices' video format requirements or by other criteria, such as bandwidth availability), they have a choice between single-rate and multi-rate multicast [1]. In the single-rate approach, the sender transmits at one fixed rate to all receivers. This rate is either suited to conform to the slowest receiver or to meet an inter-receiver fairness objective according to methods like the one presented in [2]. Multi-rate multicast is definitely more flexible and can make more efficient use of network resources. It has two basic modes. In the simulcast mode several versions (in terms of format, quality, encoding, etc., and hence bit rate) of one and the same content are available (usually no more than a dozen of versions). In the layered mode, called scalable video coding (SVC), each video is encoded into one base layer and several enhancement layers. The layers are interrelated for cumulative scalable video or are independent for non-cumulative scalable video. In this paper, we consider the cumulative layered mode, where for decoding the higher enhancement layer it is necessary that the lower ones are decoded first. Non-cumulative scalable video coding, like multiple description coding (MDC), is out of the scope of this paper. We also do not focus on methods, algorithms and protocols employing scalable video coding with the objective to provide some sort of congestion control or to maximise the utilisation of resources (by setting the optimal number of layers, the rates of the layers, etc., taking into account the network settings such as the user population and the network topology).

When network broadcast operators deliver TV channels to heterogeneous clients, they can employ simulcast or scalable video coding to cater for the various (or varying in time) user requirements. We compare these two basic methods to deliver multimedia content to a set of heterogeneous devices in terms of their bandwidth requirements in a given network setting. More specifically, we compare the SVC mode of the H.264/AVC (Advanced Video Coding) standard [3] (corresponding to MPEG-4 Part 10/SVC) to simulcast where each resolution is encoded in a single-layer bitstream, using for example H.264/AVC.

1-4244-0981-0/07/$25.00 ©2007 IEEE.

The scaling functionality was already part of the MPEG-2 standard, and although it was much studied (e.g., a similar study to ours can be found in [4]), it was almost not deployed because of its coding inefficiency. However, with the suggestion of new techniques for temporal, spatial and quality scalability coding, it was reconsidered and added as amendment to the H.264/AVC standard. Overviews of the SVC standard can be found e.g., in [5], [6] and [7].

H.264/AVC is already widely adopted and its applications range from digital television broadcast to video for mobile devices. It enjoys a large industry, business and even administrative/legislative back-up, e.g., in France [8]; it has been approved by the Digital Video Broadcast (DVB) standardisation body in Europe (in 2004) for broadcast television in Europe and many terrestrial television providers employ it throughout the world nowadays; it has been recommended by 3GPP in [9] for mobile TV deployments. The question in which mode it is preferable to deploy multi-rate video multicasting, namely in simulcast or in layered mode (SVC), is still an open issue (e.g., in [10] the authors explore its suitability in a single-layer encoding versus scalable encoding mode in a mobile video delivery setting). With this paper, we aim to contribute to this discussion in an IPTV network.

In this paper, we try to answer the question how much capacity (bandwidth) is needed with both approaches, given the network conditions and user behaviour particularities. By comparing the simulcast and the scalable approach we show the limit of superiority of the scalable mode. This we do by first constructing an exact analytical model. Due to its computational expensiveness and hence limitation in some cases, we justify the assumption that the probability distribution of the required capacity can be approximated by a Gaussian distribution. We produce results based on that assumption, which we verify by comparison with the exact model and by simulations. We also explore the impact of particular parameters through case studies.

The paper is organised as follows. First we present the assumptions of the model. Then in Section II we present the theory and simulation tools developed and in Section III we show some practical results produced with them. In Section IV we comment on the results and we draw the conclusions of this work.

B. Assumptions of the Model

We consider a TV broadcast network with N subscribers that are heterogeneous with respect to their multimedia receivers (there are numerous models imaginable – for example every user has only one device or every user has several devices and selects one of them, and many other models). The users are offered a set of K streamed TV channels that have an a priori known popularity distribution. In this paper, we represent the popularity of the different channels by a Zipf distribution [11] with parameter α (typically $0 < \alpha < 1$). Although our model can deal with any probability distribution for the popularity of the channels, we choose the Zipf distribution because it is largely accepted in the research community for modelling popularity distributions (e.g., [12], [13]). Fig. 1 shows an example network set-up (for a DSL (Digital Subscriber Line) IPTV provider).

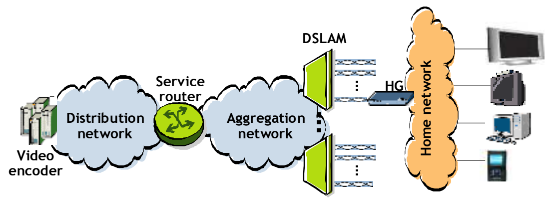


Fig. 1. Multi-rate multicast-capable IPTV distribution network.

Our model is not confined to a given network topology or underlying technology, but assumes a multi-rate multicast-capable TV distribution network.

A video server streams the bouquet of channels through a distribution network on to an aggregation network and down to the head-end customer device through an access distribution network (containing a DSL Access Module (DSLAM) and home gateway (HG) in the example in Fig. 1). We compare the capacity that is required (on the aggregation network) by the simulcast and scalable video coding approaches for delivering the streamed multicast video content in a scenario where the number of versions for simulcast or respectively layers for SVC is predefined by the supported resolutions. We assume that there are L different resolutions, determined by, for example, the video format limitations of the customer's devices (e.g., QCIF, CIF, SDTV and HDTV formats). Further in this paper, we will explain the user behaviour patterns we consider.

We denote by $R_{sim,l}$ the bit rate required for streaming a channel in version l using simulcast (for example, 0.128Mb/s, 0.384Mb/s, 1.5Mb/s and 6Mb/s for QCIF, CIF, SDTV and HDTV respectively). By $R_{svc,l}$ we denote the required bit rate in SVC mode when layer 1 until layer l of a channel are requested. The highest bit rate version in simulcast is denoted by $R_{sim,L}$ and the corresponding top-resolution bit rate in SVC is denoted by $R_{svc,L}$. It is largely known (e.g., [14], [15]) that in order to offer the same video quality with scalable and non-scalable video coding, there is some bandwidth penalty to be paid when applying scalable video coding techniques. In order to attain the same quality at all resolutions, SVC requires a higher bit rate than AVC, except for the lowest resolution (the SVC stream has an H.264/AVC base layer). We denote this excess bit rate penalty by ε, and therefore express the rate corresponding to SVC layers 1 until l by

$$\begin{aligned} R_{svc,1} &= R_{sim,1} \\ R_{svc,l} &= R_{sim,1} + \sum_{i=2}^{l}\left(R_{sim,i} - R_{sim,i-1}\right)(1+\varepsilon) \quad , \qquad l > 1 \end{aligned} \tag{1}$$

Remark that the bit rate of the SVC base layer equals that of the lowest resolution version of the simulcast versions. Although in [16] a value of at most 10% is recommended for ε, this is often hard to achieve with the current implementations. For example, in [6] the authors conclude that if an optimised SVC encoder control is applied, with spatial and quality scalability the bit rate increase relative to non-scalable H.264/AVC can be "as low as" 10%, which means that in many cases this value is surpassed. In reality, ε can easily go beyond 20 to 30%, especially for videos with complicated texture scenes and a lot of movement [15]. Notice also that we assume that the bit rate corresponding to a given resolution l is the same for every channel k ($1 \le k \le K$), i.e., $R_{sim,l}$ or $R_{svc,l}$ respectively. In reality this is often the case in an IPTV network.

II. THEORY

In this section, we present several approaches to determine the network capacity demand $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$ in the simulcast and SVC mode respectively. We use the following convention throughout this paper: random variables are highlighted in bold, unlike the concrete values the random variables assume. Remark that $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$ are random variables, since there is randomness in a user's behaviour: a user watches television or not and selects a TV channel among K available channels at one of L possible resolutions. We first define the capacity demand random variables $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$ in Section II.A, then we describe the user behaviour models in Section II.B, and after that we describe the exact approach for calculating the distributions of $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$ in Section II.C. In the following Section II.D, a Gaussian approximation to estimate the capacity demand is presented, and finally in Section II.E we describe a simulator tool we developed to verify the analytical bandwidth demand estimation.

A. *Capacity Demand*

We define the random variables $\boldsymbol{n}_{k,l}$, $1\leq k\leq K$, $1\leq l\leq L$, as the number of users watching channel k in resolution l ($\boldsymbol{n}_{k,l}$ is therefore a discrete variable and assumes integer number values). Because with simulcast every channel is encoded in L independent versions, and a channel is streamed in version l if at least one user watches that channel in resolution l, we can express $\boldsymbol{C}_{sim}$ in the following way:

$$\boldsymbol{C}_{sim} = \sum_{k=1}^{K}\sum_{l=1}^{L} R_{sim,l} 1_{\{\boldsymbol{n}_{k,l}>0\}} \quad , \tag{2}$$

where $1_{\{m\}}$ is the indicator function of the event m, expressing that there is a contribution $R_{sim,l}$ to $\boldsymbol{C}_{sim}$ for every channel k that is watched in resolution l by at least one user.

With scalable video coding, in order to watch a channel in resolution l, all layers 1 until l of that channel are needed for decoding, because they are all interrelated. Thus, layer 1 until l of channel k need to be transported if there is at least one user watching channel k in resolution l, while at the same time there is no user watching channel k in a resolution higher than l. Therefore, $\boldsymbol{C}_{svc}$ is expressed as

$$\boldsymbol{C}_{svc} = \sum_{k=1}^{K}\sum_{l=1}^{L} R_{svc,l} 1_{\{\boldsymbol{n}_{k,L}=0,\, \boldsymbol{n}_{k,L-1}=0,\, \ldots,\, \boldsymbol{n}_{k,l+1}=0, \boldsymbol{n}_{k,l}>0\}} \quad . \tag{3}$$

B. *User Behaviour Models*

A large number of user behaviour models are imaginable in a multi-rate multicast system with heterogeneous receivers. In this paper, we will consider two different of them. In user model I, we assume that every user is associated with only one resolution, determined for example by the capability of his receiver terminal. Therefore, a user can only request video content in that particular resolution (and is associated to a user group). Another possibility, which we refer to as user model II, is that every user can randomly choose the resolution in which he or she wishes to watch the selected channel among all offered resolutions.

In both models we consider, the TV channels are assumed to have independent relative probabilities $\pi_1,\ldots,\pi_K$ of being requested by an active user. These probabilities are assumed to correspond to the popularity of the channels. As mentioned earlier, we consider the Zipf distribution to model the popularity of the channels, meaning that if channel k is the k-th most popular channel of all K offered channels, then

$$\pi_k = dk^{-\alpha}\ , \quad \text{for } k=1, 2, \ldots, K\ , \tag{4}$$

where α is the Zipf distribution parameter and d is a normalisation constant ensuring that the sum of all the probabilities is 1.

For both user behaviour models, we will calculate the probability generating function of the random variables $\boldsymbol{n}_{1,1},\boldsymbol{n}_{2,1},\ldots,\boldsymbol{n}_{K,L}$, defined as

$$F(z_{k,l};\forall k,l) = E\left[\prod_{k=1}^{K}\prod_{l=1}^{L} z_{k,l}^{\ \boldsymbol{n}_{k,l}}\right] \ . \tag{5}$$

For user model I, the group of N users is divided into L fixed sets (groups). All users in set l have a terminal capable of receiving resolution l, and there are N_l users of type l such that the sum of all N_l is N. All users of set l are assumed to have the same activity grade a_l. The activity grade of a user is the probability that the user is active (i.e., that he or she watches a channel), and equals the average fraction of the time that the user watches television. We then have that the sample space corresponding to the vector of random variables $(\boldsymbol{n}_{1,1},\boldsymbol{n}_{2,1},\ldots,\boldsymbol{n}_{K,L})$ is constituted by all tuples $(n_{1,1},n_{2,1},\ldots,n_{K,L})$ for which $n_l = \sum_{k=1}^{K} n_{k,l} \leq N_l$. As a consequence, there are $\prod_{l=1}^{L}\binom{K+N_l}{K}$ elements in the sample space. The probability corresponding to a tuple $(n_{1,1},n_{2,1},\ldots,n_{K,L})$ is given by

$$\Pr\begin{bmatrix} \boldsymbol{n}_{1,1}=n_{1,1}\ ,\ldots,\boldsymbol{n}_{K,1}=n_{K,1}\ ;\ldots; \\ \boldsymbol{n}_{1,l}\ =n_{1,l}\ ,\ldots,\boldsymbol{n}_{K,l}=n_{K,l}\ ;\ldots; \\ \boldsymbol{n}_{1,L}=n_{1,L}\ ,\ldots,\boldsymbol{n}_{K,L}=n_{K,L} \end{bmatrix} = \prod_{l=1}^{L}\left[\frac{N_l\,!}{(N_l-n_l)!}(1-a_l)^{N_l-n_l}\, a_l^{\ n_l} \prod_{k=1}^{K}\frac{(\pi_k)^{n_{k,l}}}{n_{k,l}\,!}\right] \ , \tag{6}$$

which implies that the probability generating function equals

$$F(z_{k,l};\forall k,l) = \prod_{l=1}^{L}\left[1-a_l+a_l\sum_{k=1}^{K}\pi_k z_{k,l}\right]^{N_l} \ . \tag{7}$$

In user model II, each of the N users is able to watch any of the L resolutions. Resolution l is selected with probability b_l, $1\leq l\leq L$, and all users are assumed to have an activity grade of a. We now have a sample space of size $\binom{N+KL}{KL}$, which contains all the tuples for which $n=\sum_{k=1}^{K}\sum_{l=1}^{L} n_{k,l} \leq N$. The formula to calculate the probability that corresponds to an element of the vector of random variables $(\boldsymbol{n}_{1,1},\boldsymbol{n}_{2,1},\ldots,\boldsymbol{n}_{K,L})$ in user model II is

$$\Pr\begin{bmatrix} \boldsymbol{n}_{1,1}=n_{1,1},\ldots,\boldsymbol{n}_{K,1}=n_{K,1};\ldots; \\ \boldsymbol{n}_{1,l}=n_{1,l},\ldots,\boldsymbol{n}_{K,l}=n_{K,l};\ldots; \\ \boldsymbol{n}_{1,L}=n_{1,L},\ldots,\boldsymbol{n}_{K,L}=n_{K,L} \end{bmatrix} = \frac{N!}{(N-n)!}(1-a)^{N-n}a^{n}\prod_{l=1}^{L}\prod_{k=1}^{K}\frac{(\pi_k b_l)^{n_{k,l}}}{n_{k,l}!} . \qquad (8)$$

In this case, the probability generating function is of the following form:

$$F(z_{k,l};\forall k,l)=\left[1-a+a\sum_{k=1}^{K}\pi_k\sum_{l=1}^{L}b_l z_{k,l}\right]^N . \qquad (9)$$

C. Exact Solution

An obvious way to tackle the problem of calculating the exact probability mass functions of $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$ is to calculate the probability mass function corresponding to the vector of random variables $(\boldsymbol{n}_{1,1},\boldsymbol{n}_{2,1},\ldots,\boldsymbol{n}_{K,L})$, using the formulas of Section II.B, namely (6) and (8). Then the probabilities of the events considered by the indicator functions in (2) and (3) are easily obtained from this probability mass function. Notice however that this way to calculate the exact solution is not straightforward, because the number of tuples $(n_{1,1},n_{2,1},\ldots,n_{K,L})$ grows fast with the parameters of the model.

Therefore, we will describe another approach we use to calculate the exact results. Observe first that the capacity demands only depend on the number of channels of each resolution that should be transported, and not on which particular channels are requested (since we assumed that the bit rates corresponding to the different resolutions are the same for every channel k), neither on exactly how many users watch a particular channel in a certain resolution. However, all this information is taken into account first, after which this information vanishes again when working out (2) and (3), by summing the appropriate probabilities. We will exploit the fact that only the required number of channels of each resolution is important, in a divide and conquer method that will be explained below.

We define for every $l \in \{1,\ldots,L\}$ the random variables $\boldsymbol{w}_l$ and $\boldsymbol{c}_l$, where $\boldsymbol{w}_l$ represents the number of users that are watching a channel in resolution l; in the simulcast case $\boldsymbol{c}_l$ denotes the number of channels that are simultaneously watched in resolution l by the user population, while with scalable video coding $\boldsymbol{c}_l$ denotes the number of channels for which layer 1 until layer l need to be transported (but no layers higher than l). Then $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$ respectively, take the values $\sum_{l=1}^{L}c_l R_l$, with $R_l = R_{sim,l}$ and $R_l = R_{svc,l}$ respectively, with probability

$$\sum_{(w_1\ldots w_L)\in W}\begin{pmatrix}\Pr[(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_L)=(c_1,\ldots,c_L)\mid(\boldsymbol{w}_1,\ldots,\boldsymbol{w}_L)=(w_1,\ldots,w_L)] \\ *\Pr[(\boldsymbol{w}_1,\ldots,\boldsymbol{w}_L)=(w_1,\ldots,w_L)]\end{pmatrix}, \qquad (10)$$

where W is the set of all possible values the vector of random variables $(\boldsymbol{w}_1,\ldots,\boldsymbol{w}_L)$ can take. Of course, if there are some values $(c_1',\ldots,c_L')\neq(c_1,\ldots,c_L)$ for which $\sum_{l=1}^{L}c_l' R_l=\sum_{l=1}^{L}c_l R_l$, their corresponding probabilities should be added. For user model I, W contains all tuples $(w_1,\ldots,w_L)$ for which for every l, $0\le w_l\le N_l$, and we have that

$$\Pr[(\boldsymbol{w}_1,\ldots,\boldsymbol{w}_L)=(w_1,\ldots,w_L)] = \prod_{l=1}^{L}\binom{N_l}{w_l}a_l^{w_l}(1-a_l)^{N_l-w_l} . \qquad (11)$$

The set W corresponding to user model II contains all tuples $(w_1,\ldots,w_L)$ for which $0\le w\le N$, where w is defined as $w=w_1+\ldots+w_L$, and now

$$\Pr[(\boldsymbol{w}_1,\ldots,\boldsymbol{w}_L)=(w_1,\ldots,w_L)] = \binom{N}{w}a^{w}(1-a)^{N-w}w!\prod_{l=1}^{L}\frac{b_l^{w_l}}{w_l!} . \qquad (12)$$

Remark that in the simulcast case, $\Pr[(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_L)=(c_1,\ldots,c_L)\mid(\boldsymbol{w}_1,\ldots,\boldsymbol{w}_L)=(w_1,\ldots,w_L)]$ in (10) reduces to $\prod_{l=1}^{L}\Pr[\boldsymbol{c}_l=c_l\mid\boldsymbol{w}_l=w_l]$, because for simulcast whether a channel is streamed in resolution l or not depends only on the number of users watching that channel in resolution l, and not on whether or not it is also watched in another resolution. For calculating $\Pr[\boldsymbol{c}_l=c_l\mid\boldsymbol{w}_l=w_l]$, for every l, in the simulcast case, and $\Pr[(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_L)=(c_1,\ldots,c_L)\mid(\boldsymbol{w}_1,\ldots,\boldsymbol{w}_L)=(w_1,\ldots,w_L)]$ in the scalable video coding case, we use the following divide and conquer method. Consider a group of M channels, whose popularity is given by the probability distribution $P_1,\ldots,P_M$, and divide this group into two disjoint non-empty groups $\mathcal{A}$ and $\mathcal{B}$. Assume that group $\mathcal{A}$ contains the channels 1 until $M^{(\mathcal{A})}$, such that group $\mathcal{B}$ contains the channels $M^{(\mathcal{A})}+1$ until M. Within its smaller group, a channel k has a probability $P_k/(P_1+\ldots+P_{M^{(\mathcal{A})}})$ of being chosen if it belongs to group $\mathcal{A}$, and a probability $P_k/(P_{M^{(\mathcal{A})}+1}+\ldots+P_M)$ if it belongs to group $\mathcal{B}$. It can be proven that for simulcast

$$\Pr[\boldsymbol{c}_l=c_l\mid\boldsymbol{w}_l=w_l]=\sum_{c_l^{(\mathcal{A})}=0}^{c_l}\sum_{w_l^{(\mathcal{A})}=0}^{w_l}\begin{pmatrix}\Pr[\boldsymbol{c}_l^{(\mathcal{A})}=c_l^{(\mathcal{A})}\mid\boldsymbol{w}_l^{(\mathcal{A})}=w_l^{(\mathcal{A})}] \\ *\Pr[\boldsymbol{c}_l^{(\mathcal{B})}=c_l-c_l^{(\mathcal{A})}\mid\boldsymbol{w}_l^{(\mathcal{B})}=w_l-w_l^{(\mathcal{A})}] \\ *\Pr[\boldsymbol{w}_l^{(\mathcal{A})}=w_l^{(\mathcal{A})},\boldsymbol{w}_l^{(\mathcal{B})}=w_l-w_l^{(\mathcal{A})}\mid\boldsymbol{w}_l=w_l]\end{pmatrix}, \qquad (13)$$

and for scalable video coding

$$\begin{aligned}&\Pr[(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_L)=(c_1,\ldots,c_L)\mid(\boldsymbol{w}_1,\ldots,\boldsymbol{w}_L)=(w_1,\ldots,w_L)]= \\ &\sum_{c_1^{(\mathcal{A})}=0}^{c_1}\cdots\sum_{c_L^{(\mathcal{A})}=0}^{c_L}\sum_{w_1^{(\mathcal{A})}=0}^{w_1}\cdots\sum_{w_L^{(\mathcal{A})}=0}^{w_L} \begin{pmatrix}\Pr[(\boldsymbol{c}_1^{(\mathcal{A})},\ldots,\boldsymbol{c}_L^{(\mathcal{A})})=(c_1^{(\mathcal{A})},\ldots,c_L^{(\mathcal{A})}) \\ \quad\mid(\boldsymbol{w}_1^{(\mathcal{A})},\ldots,\boldsymbol{w}_L^{(\mathcal{A})})=(w_1^{(\mathcal{A})},\ldots,w_L^{(\mathcal{A})})] \\ *\Pr[(\boldsymbol{c}_1^{(\mathcal{B})},\ldots,\boldsymbol{c}_L^{(\mathcal{B})})=(c_1-c_1^{(\mathcal{A})},\ldots,c_L-c_L^{(\mathcal{A})}) \\ \quad\mid(\boldsymbol{w}_1^{(\mathcal{B})},\ldots,\boldsymbol{w}_L^{(\mathcal{B})})=(w_1-w_1^{(\mathcal{A})},\ldots,w_L-w_L^{(\mathcal{A})})] \\ *\Pr[(\boldsymbol{w}_1^{(\mathcal{A})},\ldots,\boldsymbol{w}_L^{(\mathcal{A})})=(w_1^{(\mathcal{A})},\ldots,w_L^{(\mathcal{A})}), \\ \quad(\boldsymbol{w}_1^{(\mathcal{B})},\ldots,\boldsymbol{w}_L^{(\mathcal{B})})=(w_1-w_1^{(\mathcal{A})},\ldots,w_L-w_L^{(\mathcal{A})}) \\ \quad\mid(\boldsymbol{w}_1,\ldots,\boldsymbol{w}_L)=(w_1,\ldots,w_L)]\end{pmatrix} . \qquad (14)\end{aligned}$$

In these equations, the random variables $\boldsymbol{c}_l^{(\mathcal{A})}$, $\boldsymbol{c}_l^{(\mathcal{B})}$, $\boldsymbol{w}_l^{(\mathcal{A})}$ and $\boldsymbol{w}_l^{(\mathcal{B})}$ represent the same as the random variables $\boldsymbol{c}_l$ and $\boldsymbol{w}_l$, for every l, but then limited to the channels in the groups $\mathcal{A}$ and $\mathcal{B}$ respectively, whereas $\boldsymbol{c}_l$ and $\boldsymbol{w}_l$ correspond to the original group of M channels. Obviously, if $c_l^{(\mathcal{A})}$ is larger than the number of channels in group $\mathcal{A}$ or $c_l - c_l^{(\mathcal{A})}$ is larger than the number of channels in group $\mathcal{B}$, respectively, then the first, respectively second probability in (13) equals zero. Similarly, the first probability in (14) equals zero if $c_1^{(\mathcal{A})} + ... + c_L^{(\mathcal{A})}$ is larger than the number of channels in group $\mathcal{A}$, and the same happens with the second probability in (14) if $(c_1 + ... + c_L) - (c_1^{(\mathcal{A})} + ... + c_L^{(\mathcal{A})})$ is larger than the number of channels in group $\mathcal{B}$.

Observe from (13) and (14) that the first and second probabilities in each term in the right hand side of these equations are of a similar form as the left hand side of the respective equations. So we can calculate these probabilities in a recursive way, by splitting groups of channels into two groups until all groups contain only one channel. For a group $\mathcal{Z}$ consisting of one channel, we have that for simulcast

$$\Pr[\boldsymbol{c}_l^{(\mathcal{Z})} = c_l^{(\mathcal{Z})} \mid \boldsymbol{w}_l^{(\mathcal{Z})} = w_l^{(\mathcal{Z})}] = \begin{cases} 1 & \text{if } (w_l^{(\mathcal{Z})} > 0, c_l^{(\mathcal{Z})} = 1) \text{ or } (w_l^{(\mathcal{Z})} = c_l^{(\mathcal{Z})} = 0) \\ 0 & \text{otherwise} \end{cases}, \tag{15}$$

and for scalable video coding

$$\begin{aligned} &\Pr[(\boldsymbol{c}_1^{(\mathcal{Z})},...,\boldsymbol{c}_L^{(\mathcal{Z})}) = (c_1^{(\mathcal{Z})},...,c_L^{(\mathcal{Z})}) \\ &\quad \mid (\boldsymbol{w}_1^{(\mathcal{Z})},...,\boldsymbol{w}_L^{(\mathcal{Z})}) = (w_1^{(\mathcal{Z})},...,w_L^{(\mathcal{Z})})] = \\ &\begin{cases} 1 & \begin{array}{l} \text{if } (c_1^{(\mathcal{Z})} + ... + c_L^{(\mathcal{Z})} = 1, c_l^{(\mathcal{Z})} = 1, \\ \quad w_l^{(\mathcal{Z})} > 0, w_{l+1}^{(\mathcal{Z})} = ... = w_L^{(\mathcal{Z})} = 0) \\ \text{or } ((c_1^{(\mathcal{Z})},...,c_L^{(\mathcal{Z})}) = (w_1^{(\mathcal{Z})},...,w_L^{(\mathcal{Z})}) = (0,...,0)) \end{array} \\ 0 & \text{otherwise} \end{cases} \end{aligned}. \tag{16}$$

Concerning the third probability in the terms of the right hand side of equations (13) and (14), they are easily calculated from the popularity distribution of the channels (in the original group $\mathcal{A} \cup \mathcal{B}$), thus for simulcast, we have that

$$\begin{aligned} &\Pr[\boldsymbol{w}_l^{(\mathcal{A})} = w_l^{(\mathcal{A})}, \boldsymbol{w}_l^{(\mathcal{B})} = w_l - w_l^{(\mathcal{A})} \mid \boldsymbol{w}_l = w_l] = \\ &\binom{w_l}{w_l^{(\mathcal{A})}} (P_1 + ... + P_{M^{(\mathcal{A})}})^{w_l^{(\mathcal{A})}} (P_{M^{(\mathcal{A})}+1} + ... + P_M)^{w_l - w_l^{(\mathcal{A})}}, \end{aligned} \tag{17}$$

and for scalable video coding

$$\begin{aligned} &\Pr[(\boldsymbol{w}_1^{(\mathcal{A})},...,\boldsymbol{w}_L^{(\mathcal{A})}) = (w_1^{(\mathcal{A})},...,w_L^{(\mathcal{A})}), \\ &\quad (\boldsymbol{w}_1^{(\mathcal{B})},...,\boldsymbol{w}_L^{(\mathcal{B})}) = (w_1 - w_1^{(\mathcal{A})},...,w_L - w_L^{(\mathcal{A})}) \\ &\quad \mid (\boldsymbol{w}_1,...,\boldsymbol{w}_L) = (w_1,...,w_L)] \\ &= \prod_{l=1}^{L} \binom{w_l}{w_l^{(\mathcal{A})}} \left(\begin{array}{l} (P_1 + ... + P_{M^{(\mathcal{A})}})^{w_l^{(\mathcal{A})}} \\ *(P_{M^{(\mathcal{A})}+1} + ... + P_M)^{w_l - w_l^{(\mathcal{A})}} \end{array} \right) \end{aligned}. \tag{18}$$

Each time a group of channels is split, factors similar to these probabilities appear, which are calculated in an analogous way by considering the popularity of the channels in the group that is split.

Obviously, we start the recursive scheme with $M=K$ and $(P_1,\ldots,P_K) = (\pi_1,\ldots,\pi_K)$.

Although the divide and conquer method described above allows us to obtain results for larger parameter values than the straightforward method, the number of conditional probabilities that needs to be calculated using the divide and conquer method still grows with the model parameters, especially in the scalable video coding case:

– Simulcast: $(K+1)(N_m+L)$, where $N_m=\max\{N_1,\ldots,N_L\}$ for user model I, and $(K+1)(N+1)$ for user model II,

– Scalable video coding: $\binom{L+K}{L}\prod_{l=1}^{L}(N_l+1)$ for user model I, and $\binom{L+K}{L}\binom{L+N}{L}$ for user model II.

Therefore, a Gaussian approximation method will be developed in the next section to approximate the distributions of $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$ in case of large input parameters.

D. Gaussian Approximation

To circumvent the limitations of the exact method, we consider the Gaussian distribution approximation approach, similarly as in [17] and [18]. In this approach, the random variables $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$ are assumed to have a Gaussian distribution. This approximation is more appropriate for larger values of the input parameters according to the law of large numbers [19] – in cases where the exact approach can be computationally expensive. In what follows, we present formulas to calculate the average and the variance of the random variables $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$ and in this way they are completely characterised under the assumption of a Gaussian distribution.

The average and the second moment of $\boldsymbol{C}_{sim}$ are given by

$$E\left[\boldsymbol{C}_{sim}\right] = \sum_{k=1}^{K}\sum_{l=1}^{L} R_{sim,l} \Pr\left[\boldsymbol{n}_{k,l} > 0\right] \tag{19}$$

and

$$\begin{aligned} E\left[\boldsymbol{C}_{sim}^2\right] &= \sum_{k=1}^{K}\sum_{l=1}^{L} R_{sim,l}{}^2 \Pr\left[\boldsymbol{n}_{k,l} > 0\right] + \\ &\sum_{k_1=1}^{K}\sum_{k_2=1}^{K}\sum_{l_1=1}^{L}\sum_{l_2=1}^{L} 1_{\{(k_1,l_1)\neq(k_2,l_2)\}} R_{sim,l_1} R_{sim,l_2} \Pr\left[\boldsymbol{n}_{k_1,l_1} > 0, \boldsymbol{n}_{k_2,l_2} > 0\right]. \end{aligned} \tag{20}$$

We use the following properties to find the probabilities above

$$\Pr\left[\boldsymbol{n}_{k,l} > 0\right] = 1 - \Pr\left[\boldsymbol{n}_{k,l} = 0\right], \tag{21}$$

and

$$\begin{aligned} &\Pr\left[\boldsymbol{n}_{k_1,l_1} > 0, \boldsymbol{n}_{k_2,l_2} > 0\right] = 1 - \Pr\left[\boldsymbol{n}_{k_1,l_1} = 0\right] - \\ &\Pr\left[\boldsymbol{n}_{k_2,l_2} = 0\right] + \Pr\left[\boldsymbol{n}_{k_1,l_1} = 0, \boldsymbol{n}_{k_2,l_2} = 0\right] \end{aligned}. \tag{22}$$

Thus, in order to calculate the first and second moments of (19) and (20), we only need the probabilities that for certain sets consisting of one or two of all the $\boldsymbol{n}_{k,l}$ variables (from the vector of random variables $(\boldsymbol{n}_{1,1},\boldsymbol{n}_{2,1},\ldots,\boldsymbol{n}_{K,L})$), these variables equal 0.

Similarly, the average and the second moment of $\boldsymbol{C}_{svc}$ are given by

$$E\left[\boldsymbol{C}_{svc}\right]=\sum_{k=1}^{K}\sum_{l=1}^{L}R_{svc,l}A_{k,l} \tag{23}$$

and

$$\begin{aligned}E\left[\boldsymbol{C}_{svc}^{2}\right]&=\sum_{k=1}^{K}\sum_{l=1}^{L}R_{svc,l}{}^{2}A_{k,l}\\&+\sum_{k_1=1}^{K}\sum_{k_2=1}^{K}\sum_{l_1=1}^{L}\sum_{l_2=1}^{L}1_{\{k_1\neq k_2\}}R_{svc,l_1}R_{svc,l_2}B_{k_1,l_1,k_2,l_2}\end{aligned} \tag{24}$$

respectively with

$$\begin{aligned}A_{k,l}&=\Pr\left[\boldsymbol{n}_{k,L}=0,\boldsymbol{n}_{k,L-1}=0,\ldots,\boldsymbol{n}_{k,l+1}=0,\boldsymbol{n}_{k,l}>0\right]\\&=\Pr\left[\boldsymbol{n}_{k,L}=0,\boldsymbol{n}_{k,L-1}=0,\ldots,\boldsymbol{n}_{k,l+1}=0\right]-\\&\quad\Pr\left[\boldsymbol{n}_{k,L}=0,\boldsymbol{n}_{k,L-1}=0,\ldots,\boldsymbol{n}_{k,l+1}=0,\boldsymbol{n}_{k,l}=0\right]\end{aligned}, \tag{25}$$

and

$$\begin{aligned}B_{k_1,l_1,k_2,l_2}&=\Pr\begin{bmatrix}\boldsymbol{n}_{k_1,L}=0,\ldots,\boldsymbol{n}_{k_1,l_1+1}=0,\boldsymbol{n}_{k_1,l_1}>0;\\\boldsymbol{n}_{k_2,L}=0,\ldots,\boldsymbol{n}_{k_2,l_2+1}=0,\boldsymbol{n}_{k_2,l_2}>0\end{bmatrix}\\&=\Pr\begin{bmatrix}\boldsymbol{n}_{k_1,L}=0,\ldots,\boldsymbol{n}_{k_1,l_1+1}=0;\\\boldsymbol{n}_{k_2,L}=0,\ldots,\boldsymbol{n}_{k_2,l_2+1}=0\end{bmatrix}\\&-\Pr\begin{bmatrix}\boldsymbol{n}_{k_1,L}=0,\ldots,\boldsymbol{n}_{k_1,l_1+1}=0,\boldsymbol{n}_{k_1,l_1}=0;\\\boldsymbol{n}_{k_2,L}=0,\ldots,\boldsymbol{n}_{k_2,l_2+1}=0\end{bmatrix}\\&-\Pr\begin{bmatrix}\boldsymbol{n}_{k_1,L}=0,\ldots,\boldsymbol{n}_{k_1,l_1+1}=0\quad;\\\boldsymbol{n}_{k_2,L}=0,\ldots,\boldsymbol{n}_{k_2,l_2+1}=0,\boldsymbol{n}_{k_2,l_2}=0\end{bmatrix}\\&+\Pr\begin{bmatrix}\boldsymbol{n}_{k_1,L}=0,\ldots,\boldsymbol{n}_{k_1,l_1+1}=0,\boldsymbol{n}_{k_1,l_1}=0;\\\boldsymbol{n}_{k_2,L}=0,\ldots,\boldsymbol{n}_{k_2,l_2+1}=0,\boldsymbol{n}_{k_2,l_2}=0\end{bmatrix}\end{aligned}. \tag{26}$$

Again, remark that we only need the probabilities that for certain sets consisting of $\boldsymbol{n}_{k,l}$ variables, these variables equal 0, in order to calculate the first and second moments in (23) and (24).

The easiest way to calculate the probability that all variables in certain sets of $\boldsymbol{n}_{k,l}$ variables equal zero, is from the probability generating functions in (6) and (9), by putting the $z_{k,l}$'s that correspond to the $n_{k,l}$'s that need to be zero equal to zero, and all other $z_{k,l}$'s equal to one.

E. Simulator

We wrote an event-driven C-based simulator that can simulate both the simulcast and SVC streaming modes and both user behaviour models we consider. According to the selected streaming mode, user behaviour model, activity grade of the users and popularity distribution of the channels, we calculate the complementary cumulative probability distribution function (CCDF), which we also refer to as the tail distribution function (TDF), over a sufficiently large number of samples (realisations) of the "momentarily" capacity demand.

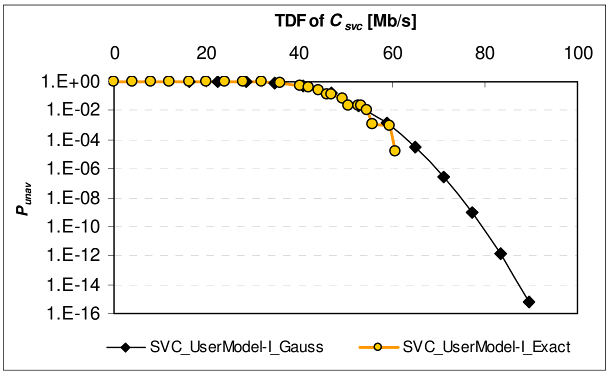
The logic implemented in the simulator is the following: it is first determined for each user (out of N) whether or not he or she is active by checking if the value generated by a uniform random number generator in the range [0,1] exceeds the value of the activity grade (a_l or a for user model I or user model II, respectively). If a user is active, a channel k is selected out of the K offered channels with probability π_k for user model I or channel k and resolution l are selected with probability $b_l\pi_k$ for user model II - again based on a random number generator. Once we know the channel and its resolution selected by a user with index n ($1\leq n\leq N$), we update a vector corresponding to a tuple $(n_{1,1},n_{2,1},\ldots,n_{K,L})$. After visiting all users, we have one instance of the random vector corresponding to $(\boldsymbol{n}_{1,1},\boldsymbol{n}_{2,1},\ldots,\boldsymbol{n}_{K,L})$. Based on a sufficiently large number of realisations $(n_{1,1},n_{2,1},\ldots,n_{K,L})$ of $(\boldsymbol{n}_{1,1},\boldsymbol{n}_{2,1},\ldots,\boldsymbol{n}_{K,L})$, the probability distributions of $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$ are calculated .

III. Results

A. Justification of the Gaussian Approximation

We first compare results obtained with the exact model to results obtained by assuming that the capacity demand variable has a Gaussian probability distribution. Results from this comparison are displayed in Fig. 2.

The upper graph of Fig. 2 shows TDFs of the required capacity for an SVC example, and the lower graph shows the same for a simulcast example. The SVC example is for K=10 channels and N=40 users, while in the simulcast example the comparison is made for model parameters as large as K=300 and N=1000. We already explained in Section II.C that the SVC case is more difficult to calculate with the exact method than the simulcast case.



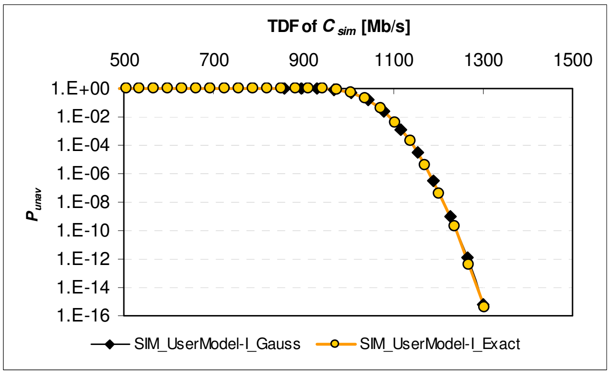


Fig. 2. Comparison of Gaussian approximation results against exact results for TDFs of $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$. The upper graph SVC parameters are L=3, K=10, α=0.6, N_1=10, N_2=20, N_3=10, a_1=a_2=a_3=0.8, and $R_{SVC,l}$ ={0.384, 1.6116, 6.5616}Mb/s. The lower graph simulcast scenario parameters are L=4, K=300, α=0.6, N_1=N_2=N_3=N_4=250, a_1=a_2=a_3=a_4=0.8 and $R_{sim,l}$ ={0.128, 0.384, 1.5, 6}Mb/s.

In both graphs, the dot-marked light curve corresponds to the exact results, while the diamond-marked dark curve shows the results obtained with the Gaussian approximation method. As expected (from the law of large numbers), for scenarios like the first one with small model parameters, the Gaussian approximation does not work satisfactorily well, while the match of results in the lower graph for a scenario with more realistic model parameters is very good. Remark that the Gaussian approximation method does not account for the fact that there is always a maximum required capacity. In the first scenario we considered, this maximum is 65.616 Mb/s, and this value is reached when all layers of all channels are provided. The probability that this maximum capacity is required is given by Pr[$\boldsymbol{C}_{svc}$>60.666 Mb/s] (60.666 Mb/s is the second largest value $\boldsymbol{C}_{svc}$ can take, and corresponds to providing all layers for 9 channels, and all but the top layer for one channel). As can be read from the curve showing the exact results, there still corresponds a rather large probability to this maximum required capacity. In the second scenario on the other hand, the maximum required capacity is 250(6+1.5+0.384+0.128) = 2003 Mb/s. From the lower graph of Fig. 2 it is clear that the probability that this maximum capacity will be needed will be much smaller than 10^{-16}, and thus also the probability by which the Gaussian approximation will predict that a capacity larger than this maximum capacity is required, will be negligible.

In order to gain further confidence that the Gaussian distribution is a good substitute for the real distributions of $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$ in scenarios with realistic (large) parameter values, we compare the Gaussian approximation results with simulation results. Note that in telecom networks, often an availability of 5 nines is required, which means that the probability that the actual capacity demand exceeds the available resources must be at most 10^{-5}. We define this probability by the term "unavailability probability" and denote it by P_{unav}. Once the TDF is known, the required capacity corresponding to a certain P_{unav} can be easily read from the TDF. In Fig. 3 we show a comparison of numerical results for the TDFs of $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$ for the two considered user behaviour models. They are obtained by the Gaussian approximation formulas and by the simulator described in Section II.D and Section II.E, respectively. Fig. 3 shows that the capacity demand is higher for the simulcast scenario than for the SVC scenario for any value of P_{unav} (for both user models). In the next section we will show that this is not always the case, which leads to the general conclusion that scalable video streaming does not always outperform simulcast in terms of required capacity. A numerical comparison of the required bandwidth (in Mb/s) for the network settings of the example in Fig. 3 is shown in Table I (made at P_{unav}=10^{-5}).

TABLE I. CAPACITY DEMAND FOR THE FOUR CASES OF FIG. 3.

BANDWIDTH DEMAND [Mb/s]	SIMUL CAST user model I	SVC user model I	SIMUL CAST user model II	SVC user model II
Gaussian approximation	1163.71	1151.85	890.97	854.36
Simulation results	1163.72	1151.65	897.19	860.84

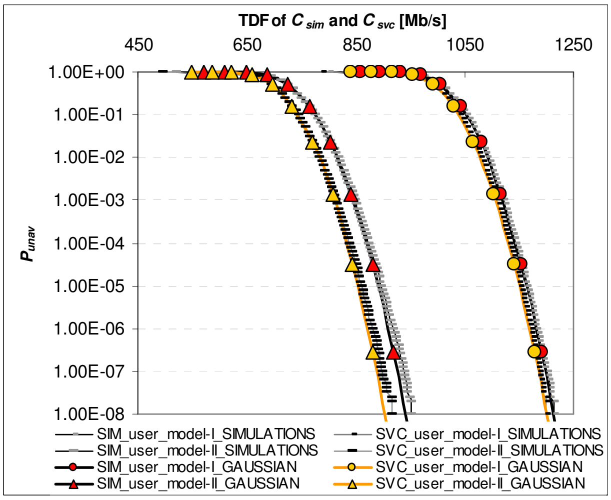


Fig. 3. Comparison of Gaussian approximation results against simulation results for TDFs of $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$. Scenario parameters are L=4, K=300, α=0.6, N_1=N_2=N_3=N_4=250 (user model I), N=1000 (user model II), a_1=a_2=a_3=a_4=0.8 (user model I), a=0.8 (user model II), b_l={0.1, 0.3, 0.5, 0.1} (user model II), and $R_{sim,l}$ ={0.128, 0.384, 1.5, 6}Mb/s, corresponding respectively to QCIF, CIF, SDTV and HDTV formats; ε = 0.15.

The correspondence of results obtained by both methods (Gaussian approximation and simulations) is good except for small values of P_{unav} for user model II.

B. *Studying the Impact of the Different Parameters Through Case Studies*

In this section, the influence of the different parameters on the required capacity for both the simulcast and SVC streaming mode is explored through case studies. We consider the required downstream capacity in the aggregation and distribution part of an IPTV network. There are four types of subscribers (i.e., L=4): mobile subscribers that receive low resolution video (QCIF) because of technology limitations (small dimensions of the devices, power consumption limitations), computer users requesting streamed video in CIF format, and TV set-top box receivers, some requesting SDTV format, and others requesting HDTV format.

In all examples to follow, we consider L=4, $R_{sim,l}$ ={0.128, 0.384, 1.5, 6}Mb/s, a_1=a_2=a_3=a_4=0.8 (user model I), a=0.8 (user model II) and b_l={0.1, 0.3, 0.5, 0.1} (user model II). In case of user behaviour model I, we evenly distribute the total number of users N over the four different resolutions, i.e., N_l = N / 4, for 1≤ l≤ 4.

1) *Impact of the SVC Bit Rate Penalty ε*

The bit rate penalty parameter ε determines the range where SVC is more beneficial than simulcast with respect to the capacity demand. There exists a limit ε (ε_{lim}) beyond which applying SVC is not beneficial anymore to saving capacity, but on the contrary, more capacity is required in the SVC delivery scenario than in the simulcast one. Fig. 4 shows the SVC capacity gain region for our considered scenario with K=300, α=0.6 and N=1000 when ε is varied from 0.1 to 0.25. From the figure it can be seen that ε_{lim} at P_{unav}=10^{-4} in user model I is around 0.16 to 0.17, and in user model II, ε_{lim} is around 0.20. Below we present an analytical formula to determine ε_{lim} exactly.

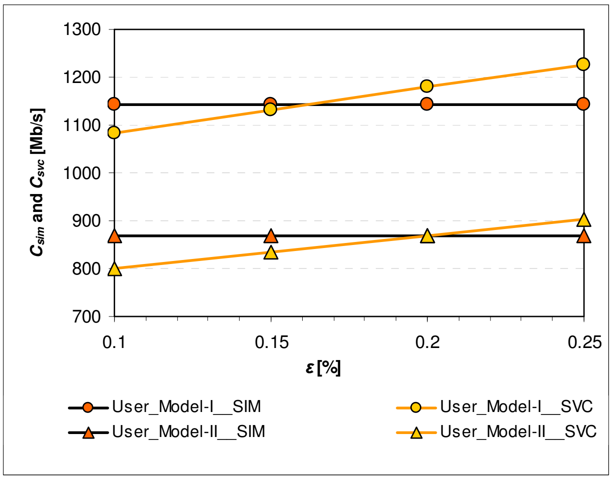


Fig. 4. Determining the limit ε beyond which there is no capacity gain at P_{unav}=10^{-4} when applying SVC streaming mode instead of simulcast. For user model I, the limit ε (ε_{lim}) equals 0.1667, for user model II it equals 0.2008 (calculated by (27)). The scenario parameters are L=4, K=300, α=0.6, N_1=N_2=N_3=N_4=250 (user model I), N=1000 (user model II), a_1=a_2=a_3=a_4=0.8 (user model I), a=0.8 (user model II), b_l={0.1, 0.3, 0.5, 0.1} (user model II), and $R_{sim,l}$={0.128, 0.384, 1.5, 6}Mb/s.

If we assume that the variance of the variables $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$ is negligible with respect to their averages (which is typically the case), then at a limit ε, denoted by ε_{lim}, the averages of $\boldsymbol{C}_{sim}$ and $\boldsymbol{C}_{svc}$, given in respectively (19) and (23), are the same, i.e., $E[\boldsymbol{C}_{sim}]= E[\boldsymbol{C}_{svc}]$. By substituting (1) in (23) we obtain that

$$\varepsilon_{lim} = \frac{\sum\limits_{k=1}^{K}\sum\limits_{l=1}^{L} R_{sim,l}\left(\Pr\left[\boldsymbol{n}_{k,l} > 0\right] - A_{k,l}\right)}{\sum\limits_{k=1}^{K}\sum\limits_{l=1}^{L}(R_{sim,l} - R_{sim,1})A_{k,l}}, \tag{27}$$

where $A_{k,l}$ is defined in (25).

If the system is saturated either by containing a huge number of users or the conditions are such that every version in simulcast and the highest resolution of every channel is requested (which means that with SVC it is needed to transport all SVC layers), $\Pr\left[\boldsymbol{n}_{k,l} > 0\right]$ tends to 1, and $A_{k,l}$ tends to 0 for lower resolutions, but tends to 1 for the highest resolutions. Thus, we arrive at ε* which is a particular case of ε_{lim}:

$$\varepsilon^{*} = \frac{\sum\limits_{l=1}^{L-1} R_{sim,l}}{(R_{sim,L} - R_{sim,1})} \quad . \tag{28}$$

This is a simpler (and a naïve approach) variant of (27), but not generally valid.

Equation (28) is the equation traditionally used to compare the performance of layered codecs with respect to simulcasting with traditional codecs. In the scenarios considered in this paper, it turns out that this leads to a too large value for the ε_{lim} beyond which SVC brings no benefit anymore.

For the practical case considered in this section, ε*=0.3426, while the actual values (calculated by (27)) are 0.1667 for user model I and 0.2008 for user model II (as illustrated in Fig. 4).

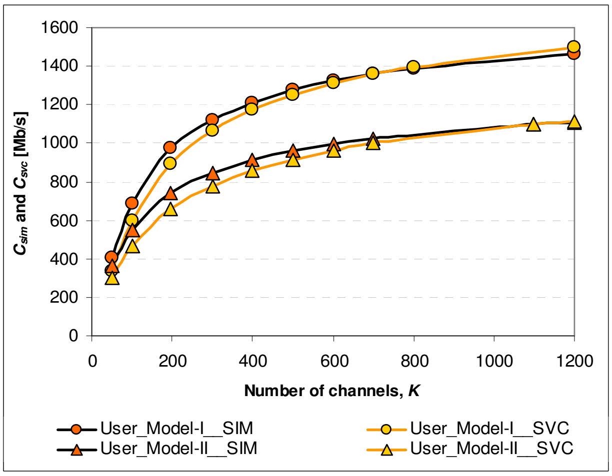


Fig. 5. Capacity demand at P_{unav}=10^{-3} for varying number of channels K. The scenario parameters are L=4, α=0.6, N_1=N_2=N_3=N_4=250 (user model I), N=1000 (user model II), a_1=a_2=a_3=a_4=0.8 (user model I), a=0.8 (user model II), b_l={0.1, 0.3, 0.5, 0.1} (user model II), $R_{sim,l}$={0.128, 0.384, 1.5, 6}Mb/s and ε = 0.1.

2) *Growing Bouquets of TV Channels - Impact of K*

For the two user models we consider and for the two modes of video streaming, i.e., simulcast and SVC, the required network bandwidth grows with increasing K, as one would expect. This is demonstrated in Fig. 5, where K is varied up to 1200 channels (with Zipf popularity probability distribution with α=0.6 for every calculation point). We again observe that SVC is superior to simulcast up to a certain value of K, beyond which SVC does not bring capacity gain anymore and is less efficient than simulcast. In the studied case, this happens around K=700 for user model I and around K=1100 for user model II. The phenomenon of the intersection point is quite complex. In a system with large N and activity grade (large active user population) it can be explained in the following way: let's compare again the averages of the simulcast and SVC cases given by (19) and (23); we always have that $R_{sim,l} \le R_{svc,l}$, while at the same time the probability calculated by (21) is always larger than or equal to the one of (25). However, for the highest most contributing layers to the capacity demand, the probabilities of (21) and (25) are close (or equal for l=L):

$$\Pr\left[\boldsymbol{n}_{k,l} > 0\right] \approx \Pr\left[\boldsymbol{n}_{k,L} = 0, \boldsymbol{n}_{k,L-1} = 0, \ldots, \boldsymbol{n}_{k,l+1} = 0, \boldsymbol{n}_{k,l} > 0\right].$$

With increasing K, the difference $\sum\limits_{i=2}^{l}\left(R_{sim,i} - R_{sim,i-1}\right)\varepsilon$ (see (1)) is added more times in SVC (i.e., in (23)), which has a higher contribution to the mean of the capacity variable when the bit rates of the highest layers are manifold larger than the ones of the lowest layers (as is the case in our example) and when ε is large. This makes that at some point of increasing K, the average required capacity with SVC surpasses the average required capacity in a simulcast scenario.

Another intuitive explanation can be that in a situation where $K >> N$ and the activity grade of users is not high, when the number of offered channels increases, many of them are not requested at all or requested in only one resolution (or a few resolutions, but not all). This is not a favourable situation for deploying SVC as by nature (and assumption in this paper in (1)) $R_{sim,l} \le R_{svc,l}$. In this way, it becomes not justified to pay the bit rate penalty ε in SVC.

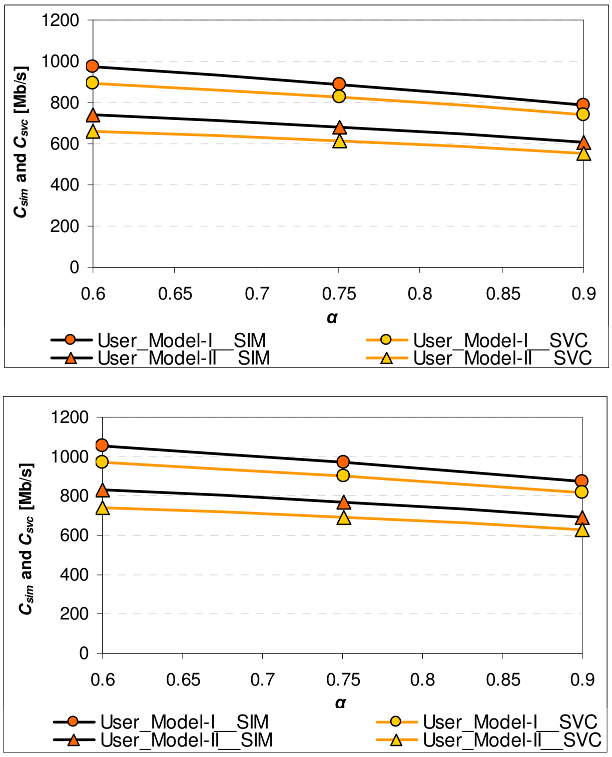


Fig. 6. Capacity demand in function of α for P_{unav} =10^{-3} (upper graph) and P_{unav} =10^{-8} (lower graph). The scenario parameters are L=4, K=300, N_1=N_2=N_3=N_4=250 (user model I), N=1000 (user model II), a_1=a_2=a_3=a_4=0.8 (user model I), a=0.8 (user model II), b_l={0.1, 0.3, 0.5, 0.1} (user model II), $R_{sim,l}$={0.128, 0.384, 1.5, 6}Mb/s and ε = 0.1.

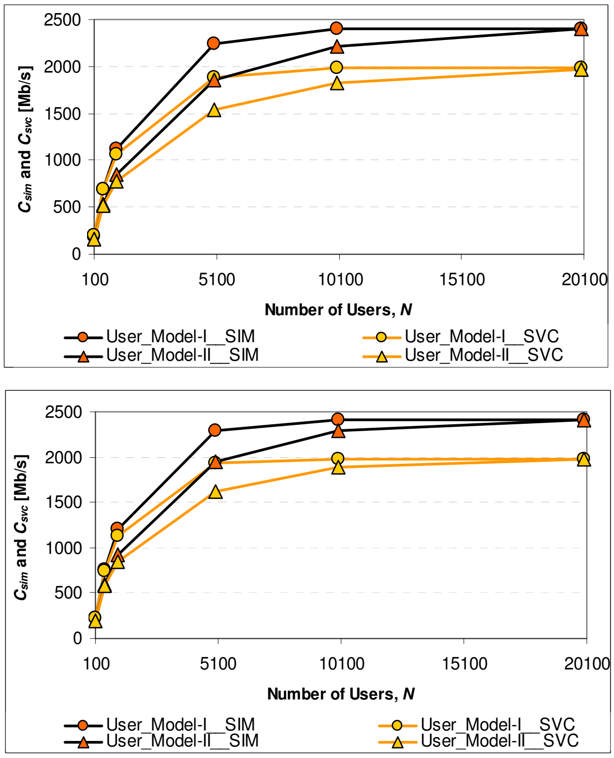


Fig. 7. Impact of N on the capacity demand when varying N up to 20000 users for P_{unav} =10^{-3} (upper graph) and P_{unav} =10^{-8} (lower graph). The scenario parameters are L=4, K=300, α=0.6, N_1=N_2=N_3=N_4=N/4 (user model I), a_1=a_2=a_3=a_4=0.8 (user model I), a=0.8 (user model II), b_l={0.1, 0.3, 0.5, 0.1} (user model II), $R_{sim,l}$={0.128, 0.384, 1.5, 6}Mb/s, and ε = 0.1.

3) Impact of the Zipf Distribution's Parameter α

The impact of α, the parameter of the Zipf distribution, on the capacity demand is straightforward. The higher the parameter α, the more homogeneous the users' preferences are, i.e., the more unanimous the users are about which channels they prefer. This is beneficial from a network capacity dimensioning point of view, as we also concluded in our previous work [18]. Fig. 6 shows the capacity demand for the four explored cases, in function of increasing α from 0.6 to 0.9 (realistic values for practical purposes), for two values of the unavailability probability: P_{unav}=10^{-3} in the upper graph and P_{unav} =10^{-8} in the lower graph. It is seen that the capacity demand decreases monotonically with increasing α for the four considered cases and for both P_{unav} values. Under the considered parameter settings, SVC is superior to simulcast in terms of required capacity, but we must note again that this is not a general rule, as we demonstrated above (in Section III.B-1 and Section III.B-2).

4) Impact of the Number of Users N

In this section we study the impact of an increasing number of users N on the required capacity. This can give, for example, some insight on the choice of the subscriber group size in the distribution part of the network. In Fig. 7 we show the increase in capacity demand as a function of N (varied up to 20000 users) for the four considered cases at P_{unav}=10^{-3} in the upper graph and P_{unav}=10^{-8} in the lower graph.

With this example we see the limitation of the Gaussian approximation.

Some of the values for the capacity demand for N≥10000 just exceed the maximum required capacity (namely 1976.2 Mb/s in the SVC scenario and 2403.6 Mb/s in the simulcast scenario). The Gaussian approximation does not account for the fact that there exists a limit capacity and obtained values beyond it are not realistic. Therefore, in such situations we set the corresponding capacity demand value to the maximum possible one.

We see that the difference in required capacity between simulcast and SVC converges to one value. The reason for this is that with increasing N, at some point the probability that all resolutions of all channels are requested is nearly 1. The required bandwidth in the simulcast case is then the sum of all $R_{sim,l}$, multiplied by the number of channels, while in the simulcast case this is the bandwidth corresponding to the top resolution, $R_{sim,L}$, multiplied by the number of channels. So the difference between the required capacity in the simulcast and SVC mode is constant beyond a certain N, and equals $K(\sum_{l=1}^{L} R_{sim,l} - R_{svc,L}) = 427.4\,\text{Mb/s}$ in the scenario considered in Fig. 7.

IV. CONCLUSIONS

In this paper, we studied the required capacity in a multicast-enabled aggregation (and distribution) IPTV network offering a bouquet of TV channels in multiple resolutions. We considered and compared two multicast streaming modes.

With the first one all resolutions are separately offered in a different (format and bit rate) version and we refer to it as simulcast mode. With the second mode all resolutions are embedded in one (multilayered) stream produced by a scalable video codec (SVC). SVC would be more efficient in terms of bandwidth demand because with simulcast all requested resolutions for each channel in the bouquet need to be transported in parallel, while with SVC only the transport of all layers up to the highest requested resolution is needed. However, it is known that with scalable video encoding there is a bit rate penalty to pay compared to non-layered video encoding when aiming at the same quality. This makes SVC less efficient than simulcast if for most channels only one resolution is needed. Thus, one could conclude also that if for most channels only one resolution is needed, scalable video coding is less efficient than simulcast. Therefore, we explored the limit of efficiency of the SVC mode over the simulcast streaming of the TV channels in terms of capacity demand.

We developed a method to derive the exact probability distribution of the required capacity, for both the simulcast and SVC case, and for two user behaviour models. Since calculating the required capacity with the exact method results in some cases in a combinatorial explosion (especially for SVC when there are a high number of channels or a high number of users), we also developed an approximate method based on a Gaussian assumption. Via comparison with the exact model and simulations, we showed that this approximate method works well when the numbers of users and channels are large enough. After having validated the Gaussian approximation we explored with it the influence of the model parameters – the SVC bit rate penalty, the number of channels, the number of users, channel popularity. We demonstrated with realistic examples that scalable video coding does not always outperform simulcast in terms of required capacity, and we derived a formula to calculate the limit SVC bit rate penalty beyond which the SVC streaming mode is less efficient than simulcast. We differentiated this formula from the naïve approach formula for the limit bit rate penalty and showed the overestimation results the latter can lead to. In a specific example we took for the two considered user behaviour models, the limit values were found to be 16% and 20%, which can be still exceeded by state-of-the-art H.264 SVC codecs.

REFERENCES

[1] B. Li, J. Liu, "Multirate video multicast over the internet: an overview", *IEEE Network*, vol. 17, no. 1, pp. 24-29, 2003.

[2] H. Yousefi'zadeh and H. Jafarkhani, "Achieving inter-receiver fairness utilizing layered media multicast control (LMMC)", *Proceedings of the IEEE International Conference on Communications*, *ICC 2004*, vol. 2, pp. 1151-1155, Paris, France, June 2004.

[3] ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, "Joint Draft 11:Scalable Video Coding" Joint Video Team (JVT) of ISO/IEC MPEG&ITU-T VCEG, Document JVT-X201, Geneva, Switzerland, July 2007.

[4] T. Kim, M. Ammar, "A comparison of layering and stream replication video multicast schemes", *Proceedings of NOSSDAV'01*, Port Jefferson, NY, USA, June 2001.

[5] S. Wenger, Y. Wang and M. Hannuksela, "RTP payload format for H.264/SVC scalable video coding", *Journal of Zhejiang University - SCIENCE A*, vol. 7, no. 5, pp. 657-667, 2006.

[6] H. Schwarz, D. Marpe, T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard", *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 17, no. 9, pp. 1103-1120, 2007.

[7] H. Sun, A. Vetro, J. Xin, "An overview of scalable video streaming", *Wireless Communications and Mobile Computing*, vol. 7, no. 2, pp. 159-172, 2007.

[8] http://www.opendivx.fr/tutoriaux/?rub=video&part=codec-h264

[9] 3GPP TS 26.234. "Transparent end-to-end Packet-switched Streaming Service (PSS); Protocols and codecs (Release 6)", April 2006.

[10] T. Schierl, T. Stockhammer, T. Wiegand, "Mobile video transmission using scalable video coding", *IEEE Transactions on Circuits and Systems for Video Technology, vol.* 17, no. 9, pp. 1204-1217, 2007.

[11] G. K. Zipf, "Selected studies and the principle of relative frequency in language", Harvard University Press, Cambridge, MA, 1932.

[12] N. Sinha, R. Oz and S. V. Vasudevan, "The statistics of switched broadcast", *Proceedings of the SCTE 2005 Conference on Emerging Technologies*, Tampa, FL, USA, January 2005.

[13] C. Heuck, "Benefits and limitations of hybrid (broadcast/mobile) networks", *Proceedings of the 14th IST Mobile & Wireless Communications Summit*, Dresden, Germany, June 2005.

[14] M. Wien, H. Schwarz, T. Oelbaum, "Performance analysis of SVC", *IEEE Transactions on Circuits and Systems for Video Technology, vol.* 17, no. 9, pp. 1194-1203, 2007.

[15] I. Amonou, N. Cammas, S. Kervadec and S. Pateux, V. Bottreau, E. François, J. Vierou, "On SVC profile A", Joint Video Team of ITU-T VCEG and ISO/IEC MPEG, Doc. JVT-V056, Marrakech, Morocco, January 2007 – available on-line at http://ftp3.itu.ch/av-arch/jvt-site/2007_01_Marrakech/.

[16] ISO/IEC JTC1/SC29/WG11 N6880 "Applications and requirements for scalable video coding", JVT document, Hongkong, China, January 2005.

[17] Z. Avramova, D. De Vleeschauwer, S. Wittevrongel and H. Bruneel, "Models to estimate the unicast and multicast resource demand for a bouquet of IP-transported TV channels", *Proceedings of the 2006 International Workshop "Towards the QoS Internet", To-QoS'2006,* pp. 137-148, Coimbra, Portugal, May 2006.

[18] Z. Avramova, D. De Vleeschauwer, S. Wittevrongel and H. Bruneel, "Dimensioning multicast-enabled networks for IP-transported TV channels", *Proceedings of the International Teletraffic Congress*, *ITC20,* pp. 6-17, Ottawa, Canada, June 2007.

[19] R. Yates, D. Goodman, "Probability and stochastic processes", pp. 247-255, John Wiley & Sons, New York, USA, 1999.