IFIP Networking 2024 1571001675
2024 IFIP Networking Conference (IFIP Networking)

Glancing at Extended Reality: an empirical model of 3D Animated XR Data Traffic

Tiziana Cattai*, Luca Mastrandrea, Alessandro Priviero, Gaetano Scarano*, Stefania Colonnese*
Dept. of Information Engineering, Electronics and Telecommunications
Sapienza University,Rome, Italy
*name.surname@uniroma1.it

***Abstract*—Several elements in the design of next generation networks, such as user profiling and network slicing, are based on precise models of the traffic load. In this context, recent research has investigated various video traffic classes, while traffic related to extended reality (XR) services remains unexplored. In this paper, we propose an original empirical model of 3D animated XR data, derived by encoding real point clouds with a standard compliant codec. Our proposed approach spans different temporal scales, from minutes to milliseconds aiming to measure different phenomena. Indeed, we firstly analyze the packet size distribution at the lower time scale, and we identify that it is well approximated by heavy tailed Gamma distribution. Then, we demonstrate how this finding can be integrated to model phenomena at the application layer time scale. Specifically, we show how a general semi-hidden Markov model can be used to capture the dynamics of the service session over time as well as the users behaviours. We demonstrate the use of the model by different examples. Taken together, our model results able to capture fine and coarse grained behaviour in 3D XR traffic.**

***Index Terms*—eXtended Reality (XR), Markov source, Traffic model**

I. Introduction

Traffic modeling is crucial in the design of next generation network [2], and it has strong impact in several problems, such as user profiling, network slicing, caching strategies, and cost-revenue analysis. The advancements of multimedia source models [19] simplify the creation of experimental test environments [22] in the development of multimedia vehicular networks [23] . Extended reality (XR) traffic varies according to the application and the type of encoded data. The kind of data mostly used in XR are 3D volumetric data, known are Point Clouds (PCs).

S. Colonnese's work was partially supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on "Telecommunications of the Future" (PE00000001 - program "RESTART"). L. Mastrandrea's, A. Priviero's work was partially supported by the European Union's Horizon HADEA research and innovation program under grant Agreement 101092851 XR2LEARN project.

ISBN 978-3-903176-63-8 ©2024 IFIP

Extended reality traffic modeling presents unique challenges, [11] owing to its broad spectrum of data types and services, including gaming traffic [6].

Existing literature has primarily focused on virtual reality data, notably traffic generated by gaming services [11]. Conversely, there is a gap in explicitly modeling 3D volumetric XR data, particularly point clouds. This kind of data introduces unprecedented challenges for next-generation networks [18] in terms of high throughput [15], [16] and stringent requirements for delay and jitter [17].

Herein, we propose an XR traffic model that is innovative because it fills a gap in modeling XR point cloud data, offering a versatile and precise model for XR traffic across different time scales. Our approach stems from the fact that in XR services the transmitted data depends both on the user state and on the encoded media type. In case of point cloud streaming, the actual transmitted traffic depends on the encoded data source features but also on application layer rate adaptation to react to the actual network quality of services [5]. Conversely, a model suitable for gaming traffic should account of user actions [6]: the user travels through different states, corresponding to different activities, like being idle, scrolling the extended reality view, or changing instantaneously the viewpoint. The state changes occur at the time scale of the user's behaviour, and so does the generated associated download traffic. For a given user state, encoded data packets are randomly generated, and the statistical properties of the packet data length change accordingly to the user's state. For instance, when the user is in the idle state, the traffic reduces to low bitrate, whereas the rate increases significantly in the scrolling state, when the user rapidly refreshes the rendered view requiring large amount of innovation data per seconds.

In a nutshell, herein we propose a model of traffic generated in XR volumetric video services, where point cloud data are encoded. The main contribution of this paper are

- we show novel results on the statistical distri-

978-3-903176-63-8/24/$31.00 ©2024 IEEE

bution of geometric and photometric XR data packets at different time scales, extending state-of-the-art results on video traffic;
- we introduce a general semi-hidden markov model accounting for different rate-adaptation phenomena, ranging from network congestion to user behaviour; the model can serve in network design procedures;
- we provide a set of encoded traffic traces together with an XR source traffic generator, to be used within network simulators of XR services.

To account for the complexity of XR data sources, we utilize a two-tier model, which considers the different time scales of XR data generation, which correspond to user behavior, network quality of service, and type of transmitted data. This is sketched in Fig.8. The user changes its state, on a time scale of the order of minutes [6]; when the user remains in a given state, several independently decodable data packets (typically referred to as Group of Pictures, or GOPs) are encoded. The rate of the GOP can be adapted to account for the actual network quality of service [5]. Each GOP is built up by individual frames, i.e. encoded data referring to different time points; in turn, each frame is encoded using one or more slices. The proposed novel model has a two-tier structure that accounts for both the data packet generation and the changes due to application layer reactions or user behaviour. The model firstly statistically characterizes the encoded XR data traffic at a fine grain temporal scale, and then represents the traffic fluctuations at a coarse-grain time scale by hidden (semi) Markov models.

The organization of the paper is as follows. Sec. II reviews the state-of-the-art. In Sec.III, we introduce a fine-grain description of the traffic, addressing the lower layer time scale; in Sec.IV, we present a coarse-grain description of the traffic, addressing the higher layer time scale. We report numerical experiments obtained using the model result in Sec.V; Sec.VII concludes the paper.

II. Related Work

The literature on XR spans different kinds of applications, such as mixed reality streaming, 3D scene understanding, showcasing novel acquisition and compression techniques in real-world scenarios.

Point clouds are a set of data points lying in the 3D space representing tridimensional shapes or objects. They are widely used in extended reality to represent detailed and accurate 3D models of scenarios [1]. In this context, the quality of point clouds is a crucial aspect to determine their actual usability. Indeed, the study in [12] explores how the quality of point clouds and the viewer's distance from those objects affect user experience in mixed reality environments. Novel quality of experience (QoE) metrics, emphasizing those factors together with subjective perception [29], could be developed.

Innovative methods have been introduced [13] to understand and interpret dynamic scenes from 2D images to 3D point clouds, focusing on the motion and flow. They propose novel information organization techniques and a new definition of motion, by leveraging deep learning and attention mechanisms in order to improve scene flow estimation in complex environments. Another important challenge is related to efficiently compress color information in dynamic 3D point clouds. One possible solution identified in [7] is to using affine transformations to reduce the amount of color data necessary to represent a point cloud without visual quality degradation.

In [8] the authors present a comprehensive analysis of mobile-app traffic patterns using Markov models to predict future traffic behavior, whereas Markov chains are introduced in [9] for network traffic prediction at various granularities. This approach aims to improve the accuracy of traffic forecasts, supporting network planning and optimization efforts in the face of varying user demands. In [10] an in depth XR traffic model is presented, referring to the specific case of video data enriched by enhanced information. Contrary to existing approaches that primarily focus on mixed/augmented reality data from video traffic, our model directly describes the specific characteristics of 3D volumetric video data of traffic generated by point cloud encoding.

III. Fine grain XR traffic model: the encoded semantic units

In this Section, we aim to empirically characterize the traffic source at a semantic unit time scale. To this aim, we collected encoded data features from 3D volumetric animated data, using a standard compliant encoder. We perform the histogram analysis in order to have compact representation of the source at a semantic unit time scale.

Since XR services usually adopt media refresh at 60 frame per second (fps), a semantic unit corresponding to an entire representation of the volumetric video data is generated every $1/60$ s. Therefore, the encoder produces packets at a temporal scale that is of the order of ten milliseconds. To infer the packet size statistical characteristics, we capitalize on numerical experiments carried out using the V-PCC point cloud encoder developed within the MPEG standardization framework. In the next section we will leverage these results to statistically characterize the single states of a higher layer model referring to larger time scales. Under these premises, the scope of the experiment is

to statistically characterize the traffic load generated by a 3D animated source based on the distribution of the sizes of the semantic units at the output of the V-PCC encoder. By doing so, we do not delve into the delays introduced by the communication channel, depending on the network status and other factors that come into play beyond the V-PCC encoder. This assumption allows to reuse the proposed encoded source model within the network design, as described in Section V.

We consider two humanoids dynamic point clouds (PCs) in the dataset [24], namely the Longdress and the Soldier sequences, which have $N = 765821$ and $N = 1059810$ vertices, respectively. Those PCs have been chosen because they have real human characters, meaning that the content is acquired from reality and not artificially generated by computer graphic techniques. This kind of data is expected to become predominant in a number of XR services, including training, telemedicine or shopping, for instance. Each PC is captured at $f_0 = 30fps$ and each sequence of PCs is $T = 10s$ long. The coordinates of each PC vertex are quantized over a $1024 \times 1024 \times 1024$ voxels cube. The attributes of the quantized vertices are the RGB components of the surface color.

We encoded the PC sequences using the TMC2 implementation of the V-PCC encoder [25], [26]. It projects each point cloud into three bidimensional maps, namely the texture, geometry and occupancy maps. Those maps are then encoded using a bidimensional hybrid transform encoder. The Group of Picture (GOP) structure of the bidimensional hybrid encoder, that is the set of encoded maps initiating with an Intra frame for random access purposes, corresponds to a group of PCs that can be independently decoded.

In our analyses, we consider three test configurations, that differ for the period among Intra encoded frames: All-Intra (AI), where all the frames are encoded with no reference to the past; Low-Delay (LD), Random Access (RA). Those configurations were tested under 5 different rates. The rate was controlled by assigning two different Quantization Parameters (QPs), one for the Attributes and one for the Geometry Map, and the Occupancy Precision Parameter, which plays a similar role for the Occupancy Map. Table I summarizes the rate control parameters used in the simulations, namely the Texture Quantization Parameter (QP), Geometry QP and Occupancy Precision Parameter (PP) .

For the different configurations, the GOP has different size. The All-Intra case for the Texture, Geometry and Occupancy maps has GOP dimensions $N_{\text{GOP}}^{(T)} = N_{\text{GOP}}^{(G)} = N_{\text{GOP}}^{(O)} = 1$.

Both the Low Delay[1] and Random Access configurations have GOP dimensions $N_{\text{GOP}}^{(T)} = N_{\text{GOP}}^{(G)} = N_{\text{GOP}}^{(O)} = 16$.The number of Slices per Frame (SpF) is identical for the All-Intra, Low Delay and Random Access configurations: $N_{\text{SpF}}^{(T)} = 2$, $N_{\text{SpF}}^{(O)} = 2$ and $N_{\text{SpF}}^{(G)} = 1$.

TABLE I
ENCODING PARAMETERS: TEXTURE QUANTIZATION PARAMETER (QP), GEOMETRY QP AND OCCUPANCY PRECISION PARAMETER (PP).

Parameter	***Q1***	***Q2***	***Q3***	***Q4***	***Q5***
Texture QP	32	28	24	20	16
Geometry QP	42	37	32	27	22
Occupancy PP	4	4	4	4	2

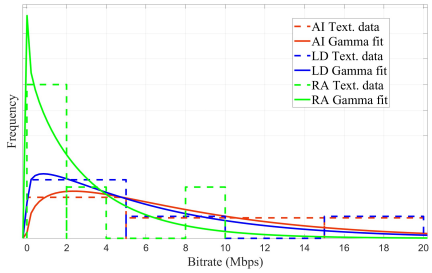

Fig. 1. Histogram of the Texture map GOP sizes observed at a GOP layer under AI, RA, LD encoding conditions (dashed lines) and the relative Gamma Distribution fitting (solid lines).

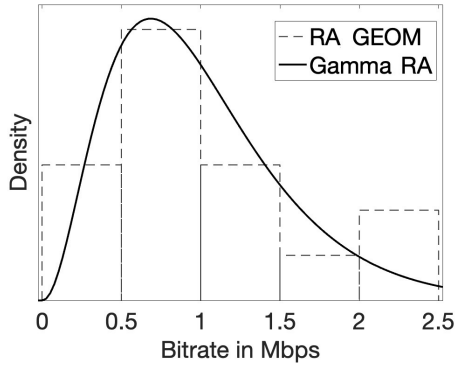

Fig. 2. Histogram of the Geometry map encoding rate observed at a GOP layer under RA encoding condition (dashed lines) and the relative Gamma Distribution fitting (solid lines).

From the experiments, we extracted the size in bits of data packet referring to both the GOPs and the Slices. The GOP data packets refer to the GOP

[1]For the Low Delay case, the term GOP is here used to denote the set of frames encoded after the first Intra frame. The actual GOP duration results from the length of the encoded PC sequence. For the sake of comparison, we have encoded sets of 16 consecutive frames.

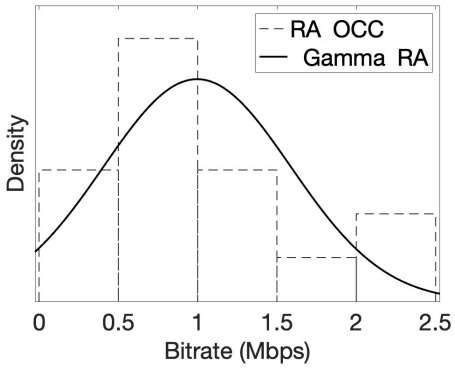

Fig. 3. Histogram of the Occupancy map encoding rate observed at a GOP layer under RA encoding condition (dashed lines) and the relative Gamma Distribution fitting (solid lines).

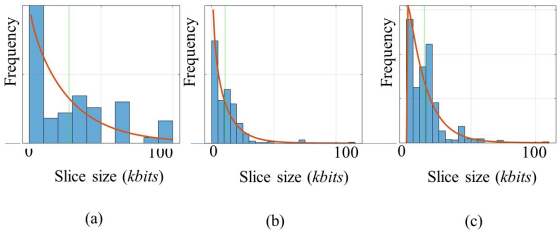

Fig. 4. Histogram of the Slice size for the Texture Map data for the three configurations represented by the blue bars with the relative Gamma Distribution fitting (red solid lines): (a) All Intra (AI), (b) Random Access (RA), (c) Low-Delay (LD).

time scale, i.e. they result from the aggregation of the encoded bits corresponding to a random access unity spanning $T_{GOP}^{(T)} = N_{\text{GOP}}^{(T)}/f_0$. In the AI case the time span of a GOP is $T_{GOP} = 0.07s$, whereas for the RA case $T_{GOP} = 0.53s$. In the LD case, the aggregate GOP data length refers to $T_{GOP} = 0.07s$.

Fig. 1 represents the histogram of the GOP sizes for the Texture Map for the three AI, RA, LD configurations, as computed over all the experiments with different qualities. We recognize that all the three configurations give rise to heavy tailed data histograms. Besides, we report the Gamma fitting of the histograms, which provides a coarse and fast statistical characterization of the XR data.

For comparison sake, we represent in Fig.2 the Gamma fitting for the case of the Geometry maps, in the Random Access encoding configurations. We recognize the wide variability of the GOP size, which is well represented by the wide, heavy-tailed distribution. This occurs because the Geometry map is basically a depth map, and its behavior is similar to that of texture maps. The same behavior for the Geometry is observed for different encoding configurations (LD, AI).

For the Occupancy maps, a wide variability of the GOP size is observed in Fig.3. Still, a wide, heavy-tailed distribution fitting can be applied, although the approximation is less tight than in the geometry case. Similar histograms are observed in the LD and AI encoding conditions. This has a two-fold motivation: firstly, the Occupancy map is a binary image, with different rate-distortion characteristics with respect to the Texture and Geometry map, definitely similar to natural video. Besides, the actual GOP size changes significantly with the rate control parameter (PP). The statistical characterization at the GOP time scale is useful for services where the transmitted data chunks correspond to a playout time of the order of a few seconds, as it occurs in streaming or gaming services. Different services, such as conference services, transmit data fragments corresponding to a smaller time scale.

With these observations, we move towards statistical characterization of the source at a Slice time scale, which is of the order of tenth of milliseconds. Fig.4 reports the histogram of the packet sizes at a Slice level, for the Texture Map data for the three configurations. The histograms encompass Intra encoded, Predicted and Bi-directionally predicted slices depending on the encoding setting. Due to the intrinsically different packet size of Intra and predicted slices, the histograms are heavy tailed. This condition is again approximately captured by Gamma distribution fitting (red line in Fig.4. Thereby, the XR source at a slice time level is well approximated by a Gamma distribution.

For the sake of concreteness, in Table II and Table III we report the mean and variance of the Slice sizes and the (α, β) parameters of the fitting Gamma distribution for different encoding conditions and qualities.

To sum up, we show in this section that the packet sizes are strongly variable: some data packets are significantly larger than others due to the highly informative nature of point cloud data, which translates into compression difficulties. In this intricate context, we show that at the packet time scale, the heavy tailed Gamma distribution provides an accurate data model.

IV. Coarse grain XR traffic model: from rate adaptation to user behaviour

With the aim to build the coarse grain traffic model, we adopt a hidden semi-Markov model to capture the dynamics of the service session over time. This model is able to represent crucial aspects as rate adaptation, occurring at a time scale of seconds, or as users behaviour, observed at a time scale of ten seconds or higher.

In hidden semi-Markov models, the states $\mathcal{S}_m$, $m = 0, \cdots M-1$ are visited during the

TABLE II
POINT CLOUD SOLDIER [24]: SLICE SIZE MEAN AND STANDARD DEVIATION (μ, σ), AND GAMMA DISTRIBUTION PARAMETERS $(k.\theta)$,
FOR DIFFERENT ENCODING CONDITIONS (ALL INTRA, RANDOM ACCESS, LOW DELAY) AND QUALITY LEVELS (Q1-Q5).

Coding mode	**Qualities**				
All Intra	***Q1***	***Q2***	***Q3***	***Q4***	***Q5***
(μ, σ)	$(27.6, 23.5)$	$(44.9, 47.0)$	$(76.1, 92.3)$	$(132.7, 170.5)$	$(236.2, 285.9)$
(σ/μ)	0.85	1.05	1.2	1.28	1.21
(k, θ)	$(0.5604, 49.3)$	$(0.6293, 71.4)$	$(0.7306, 104.2)$	$(0.8254, 160.8)$	$(1.0078, 234.4)$
Random Access	***Q1***	***Q2***	***Q3***	***Q4***	***Q5***
(μ, σ)	$(9.4, 9.3)$	$(12.7, 14.5)$	$(18.3, 25.8)$	$(33.4, 47.3)$	$(73.8, 85.0)$
(σ/μ)	0.99	1.14	1.41	1.42	1.15
(k, θ)	$(0.7754, 12.1)$	$(0.8798, 14.4)$	$(1.1144, 16.4)$	$(1.4681, 22.8)$	$(2.4189, 30.5)$
Low delay	***Q1***	***Q2***	***Q3***	***Q4***	***Q5***
(μ, σ)	$(11.6, 9.2)$	$(16.9, 15.0)$	$(27.3, 28.6)$	$(55.2, 57.9)$	$(123.8, 118.8)$
(σ/μ)	0.79	0.89	1.05	1.05	0.96
(k, θ)	$(0.9155, 12.7)$	$(1.2774, 13.2)$	$(1.6621, 16.4)$	$(1.7023, 32.4)$	$(1.7206, 71.9)$

TABLE III
POINT CLOUD LONGDRESS [24]: SLICE SIZE MEAN AND STANDARD DEVIATION (μ, σ), AND GAMMA DISTRIBUTION PARAMETERS $(k.\theta)$,
FOR DIFFERENT ENCODING CONDITIONS (ALL INTRA, RANDOM ACCESS, LOW DELAY) AND QUALITY LEVELS (Q1-Q5).

Coding mode	**Qualities**				
All Intra	***Q1***	***Q2***	***Q3***	***Q4***	***Q5***
(μ, σ)	$(28.0, 34.2)$	$(48.9, 69.0)$	$(86.8, 130.9)$	$(162.0, 239.4)$	$(293.3, 404.3)$
(σ/μ)	1.22	1.41	1.51	1.48	1.38
(k, θ)	$(0.5465, 51.2)$	$(0.559, 87.4)$	$(0.596, 145.6)$	$(0.6298, 257.2)$	$(0.6921, 423.7)$
Random Access	***Q1***	***Q2***	***Q3***	***Q4***	***Q5***
(μ, σ)	$(12.7, 14.7)$	$(20.2, 27.3)$	$(34.7, 55.1)$	$(67.9, 112.4)$	$(58.2, 222.0)$
(σ/μ)	1.56	1.35	1.59	1.66	3.81
(k, θ)	$(0.7116, 17.8)$	$(0.7422, 27.2)$	$(0.7455, 46.5)$	$(0.7491, 90.7)$	$(09303, 170.0)$
Low delay	***Q1***	***Q2***	***Q3***	***Q4***	***Q5***
(μ, σ)	$(17.9, 18.1)$	$(31.2, 38.1)$	$(60.3, 82.8)$	$(126.0, 172.4)$	$(248.2, 320.2)$
(σ/μ)	1.01	1.22	1.37	1.37	1.29
(k, θ)	$(0.7207, 24.8)$	$(0.729, 42.8)$	$(0.7338, 82.1)$	$(0.7197, 175.0)$	$(0.7564, 328.1)$

service session, for time $t \in [0, T]$. Specifically, at consecutive time stamps $t_k, k = 0, 1 \cdots K-1$, a state change occurs; a sequence of states of indexes $m_{t_k}, k = 0, 1 \cdots K-1$ is hence visited. The temporal state duration $\tau_k = t_{k+1} - t_k$ may follow different probability distributions, accommodating both stationary and non-stationary models.

As far as rate adaptation is concerned, we observe that the application layer typically drives the selected rate by accounting for estimated download throughput and receiver buffer occupancy. As a rule of thumb, when harsh channel conditions result in high latency and possible decoder buffer depletion, the rate is conveniently decreased. On the contrary, during steady state, the rate is slightly and steadily increased till the channel throughput is matched. This policy is well resumed by the hidden Markov model of rate adaptation illustrated in Fig.5: the rate adaptation typically occurs at a constant temporal pace $\tau_k = \overline{\tau}$, of the order of a few seconds, depending on the quality of the communication channel; specifically, the probability of transition from one state to another depends on the adopted rate adaptation algorithm [20]. For instance, the transition probability from an initial state that is characterized by a video rate/quality might be conditioned to properties at time t_k. Those can be statistics related to the quality of services at the lower layers, such as the number of re-transmissions or the accumulated delays.

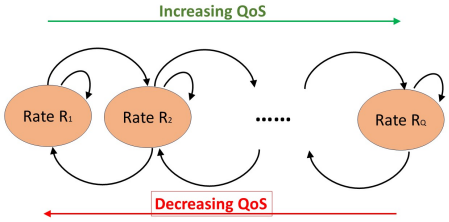

Fig. 5. Application layer rate adaptation model: in streaming services, the rate adaptation occurs at a typically constant temporal pace, of the order of few seconds.

As for user behavior modeling, we can still resort to a hidden semi-Markov model [3], which was effectively used for multiview video streaming sources. Let $\mathcal{S}_m$, $m = 0, \cdots M-1$ denote the possible states

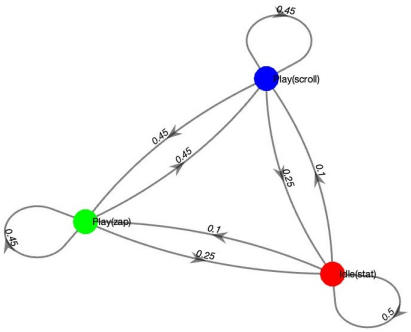

Fig. 6. Markovian model of the user activity: case of gaming user [6]. The user travels through different states, corresponding to different activities, like being idle, scrolling the extended reality view, or instantaneously changing the viewpoint.

characterizing the user behaviour (idle, zapping, interacting, and so on). We assume that state sequence is hidden semi-Markovian, where the state change is drawn at a pace determined by a random distribution: the transition, occurring at time t_k, of the users from a state $\mathcal{S}_{m_{t_k}}$ to a state $\mathcal{S}_{m_{t_{k+1}}}$ occurs with an inter-arrival time $\tau_k = t_{k+1} - t_k$ determined according to the service under concern. The transitions reflect a typical behaviour throughout an XR service session: the user visits different states, for instance, plays, or remains idle or scrolls the volumetric data. The Markov model effectively represents the dynamics of user behavior during the XR session. As for the interarrival time, Poisson distribution proved effective in modeling different processes, such as activity levels in video sequences, as well as user behaviour in multi-view video services [3]. Thereby, we herein can safely approximate the interarrival time τ_k between states transitions with Poisson distribution.

By utilizing a Markov chain, we can capture the probabilistic transitions between different user states, providing insights into user preferences and actions. The number of states M strictly depends on the service characteristics; for instance, in [6] the authors identify up to 5 playing modes for a gaming user, that could therefore be modeled by setting M equal to 5. Still, the proposed model is easily tuned to different user profiles, whether human or machine, as long as their behavior can be well approximated by a Markovian model. For example, the model can be adapted to a multi-rig camera surveillance system mounted on a UAV, where UAV tasks like hovering, charging, and being idle can be mapped to Markov chain states. For the mentioned UAV example, where the end user is a camera equipped UAV, the Markov chain states may correspond to the UAV mobility (e.g. hovering, charging, idle) and M can be set equal to 3. In applications, the Markov chain transition matrix can be suitably learned from real traffic data corresponding to different service categories [4].

V. Application of the XR Model to Next-Generation Network Scenarios

In this Section we will provide an example of how the XR traffic model can be adopted for next-generation network design. The availability of an XR traffic model enables i)identification of bandwidth and throughput needs for XR support in next-generation networks, ii) users profiling based on the estimation of the latency of the system under different condition, iii) realistic simulation of the unique requirements of XR applications, including high throughput, low latency, and the ability to handle a high density of users in both physical and virtual environments, and iv) the development of a network Digital Twin, where the model is trained to mimic the XR user load and quality of experience, definitely improving synchronization between physical and digital entities.

A. XR service example

To demonstrate the model's applicability to a wide range of XR services, we show how it can be applied to generate traffic load in gaming. The modifications needed for application to the streaming and educational frameworks will also be described.

Let us consider the simulation of a gaming user scenario, as described in Fig.7.

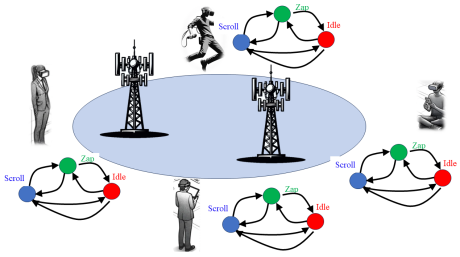

Fig. 7. Simulation scenario: each user follows a different pattern, described by a Markov model. The state changes occur at intervals drawn by a Poissonian distribution, with average values of the order of minutes. During each state, the size of the generated packets are drawn from a Gamma distributed random variable, whose parameters are state and quality dependent. The design of the allocated resource can be tested within this realistic traffic model.

This involves the definition of a synthetic traffic generator, as follows: i) a Markov process is considered, wherein transitions occur among the states Idle, Scroll, and Zapping. The transition are dictated by the semi-Markov model, e.g. they occur with Poissonian interval distribution; ii) within each state, a few (e.g. 5) distinct quality levels are available. For every state and quality level combination, a gamma distribution is employed to delineate packet size.

An example of the traffic generated by the model is plotted in the lower part of Fig.9, showing the packet size s_n for $n = 0, \cdots N-1$ for $N = 2500$ slices generated by a gaming user described by a three state Markov chain model (idle, zap, scroll), with transition matrix $P = [0.1, 0.8, 0.1, 0.150.10.75, 0.750.150.1]$. The duration of each state follows a Poissonian model, with average duration of $\Delta_i = 2000$, $\Delta_z = 2000$, $\Delta_s = 4000$ slices for the idle, zap, scroll state, which corresponds to a time interval to $T_i \approx 30s$, $T_z \approx 30s$,$T_i \approx 60s$, respectively. For the sake of clarity, we report in the higher part of the figure the relative average rate of the three idle, zap, scroll states, plotted in red, green, blue. The vertical colored bands in light red, green and blue highlight the temporal intervals corresponding to the different user states. From Fig.9 we recognize that the model flexibly accounts for the traffic generated by a dynamic XR source. The above described model captures the statistical feature of an XR source by joint characterization of the user and of the source encoded data.

When applied in the scenario in Fig.7, each packet size, generated every $\tau = 1/60$, shall be translated into an equivalent requested throughput $\theta_n = s_n/\tau$ bit per seconds. The so characterized traffic load will be used as an input to the network design simulation; for instance, for a user characterized by a physical layer spectral efficiency of η bit per second per Herz, the requested bandwidth is computed as $B_n = \theta_n/\eta$ Hz. The characterization of different users shall provide a realistic testbed for radio access network design and optimization.

B. HMM Parameter inference

The Markov chain parameters can be either a priori set to describe known users behaviours patterns, or estimated by application of the Expectation-Maximization algorithm from short observations of users behaviour, as described in [3]. Then, the generated traffic conditional to the state can be randomly drawn from a heavy tailed Gamma distribution, that effectively describes not only Texture, but also Geometry and Occupancy data. Depending on the time scale of interest, the GOP fitting model or the slice fitting model can be adopted.

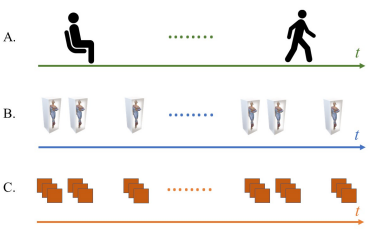

Fig. 8. The different time scales involved in XR source modeling: (a) user model (minutes); (b) rate adaptation (seconds); (c) semantic units (tenths of seconds). The proposed model addresses the different time scales.

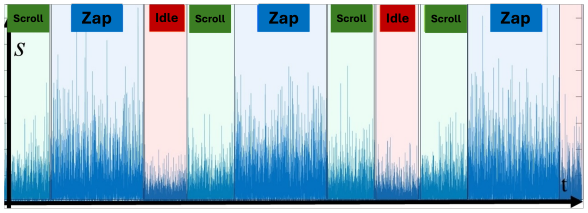

Fig. 9. Example of slice traces versus time generated from a three state Markov model of a gaming user; at each state, the slice size are drawn from a Gamma distribution of state-dependent quality.

C. Further scenarios

Let us now consider the extension of the model to different scenarios where it can be used to study orchestration and recovery mechanisms within the transmission and reception stages. This may include, but it is not limited to, adaptive streaming, group training, patrolling to mention a few.

The model can be adapted in terms of hidden Markov model states, which may vary in meaning and number; state duration (constant or random); transition probability matrix (constant or conditioned to the channel state). Besides, the model can be tuned by resorting to different probability density function (PDF) of the encoded semantic unit packet size; typical rates, characterized in terms of mean and standard deviations, shall be adapted to the service under consideration. A Table summarizing possible tuning of the model parameters to a given application is shown in Table IV.

For concreteness sake, let us consider the case of rate adaptation. In streaming services, the choice of the encoded content rate is dynamically adapted to the ongoing communication channel. The rate adaptation is client driven, and is carried out at multiples of the application layer time-slot (e.g. 2s for most streaming services [21]). Therefore, at every time-slot the quality transitions can obey to a constant Markov matrix $\mathbf{P}_Q$, or be tuned to the simulated underlying communication channel [19], in accordance to the application layer rate adaptation protocols that decrease the requested rate in case of insufficient throughput. In this setup, the packet size emitted at each state is determined by both the current state quality level.

To sum up, the semi-Markov configuration enables the generation of synthetic traffic patterns, mimicking user behavior and quality fluctuations within a networked environment. Let us observe that the state and quality level could be tuned to represent the encoder complexity and flexibility.

TABLE IV
MODEL APPLICATION TO TRAFFIC SIMULATION: FOR EACH KIND OF SERVICE, THE HIDDEN MARKOV MODEL STATES VARY IN MEANING, NUMBER, AND DURATION; THE ENCODED SEMANTIC UNIT DATA OBEY TO DIFFERENT PDFS, AND ARE CHARACTERIZED BY DIFFERENT MEAN AND STANDARD DEVIATIONS.

Service	States	M	Transition interval	Transition matrix	Packet Size PDF	Mean (Mbps)	Std.Dev. (Mbps)
Streaming	Quality levels	$5-10$	Constant	Adaptive	Gamma	$10-20$	$5-10$
Gaming	User condition	3	Poisson PDF	Constant	Truncated Gaussian	$20-100$	$10-50$
3D call	View tiles	$5-10$	Poisson PDF	Constant	Truncated Gaussian	$10-20$	$5-10$

All the aforementioned scenarios are current under investigation to develop a simulator of XR over 5G networks from the radio layer to the application layer. The model provides a theoretically grounded tool to synthesize traffic packets taking into account realistic user behaviour. Since collection of real XR traffic traces in a variety of scenarios is restrained by the still limited XR diffusion, the availability of synthetic traffic traces is relevant in developing data driven, AI based (e.g. Reinforcement Learning based) resource orchestration methods.

A few remarks are in order about the potential model limitations. Firstly, it is suited to 3D volumetric video data, represented by point clouds. Different kinds of XR data, e.g. annotated video, are not covered by the present model, and are manageable resorting to other models in the literature, e.g. [19]. Besides, the model parameters shall be tuned to the application under concern. Choices as the number of quality levels or users behaviour conditions, determining the number of states M of the Markovian model, shall be adapted to the problem under concern. The same relates to the probabilistic description of the Markov model evolution.

VI. 3D ANIMATED XR DATASET

The model leverages a semantic unit description of the data. To this aim, we release a set of XR encoded traces, and a Matlab implementation of the model at [28]. For application in a network environment, the traces provided by the Markov model source shall be complemented with information about the visual quality of the 3D animated video data. For visual quality characterization, we present the rate distortion $R(D)$ traces for the Texture and Geometry Maps. Specifically, Fig.10 plots the rate (kbps) versus the distortion, measured as the inverse of the Peak Signal to Noise Ratio, and proportional to the encoding mean square error. A given rate value $\overline{R}$ is associated to the corresponding distortion $\overline{D}$, providing a coarse measure of the data visual quality. While this representation refers to the visual characteristics of the XR data, the user quality of experience also depends on delivery delays, jitter and packet losses. These factors shall be taken into account by injecting the traffic load into the simulation scenario and then accounting for the latency at the receiver side [27].

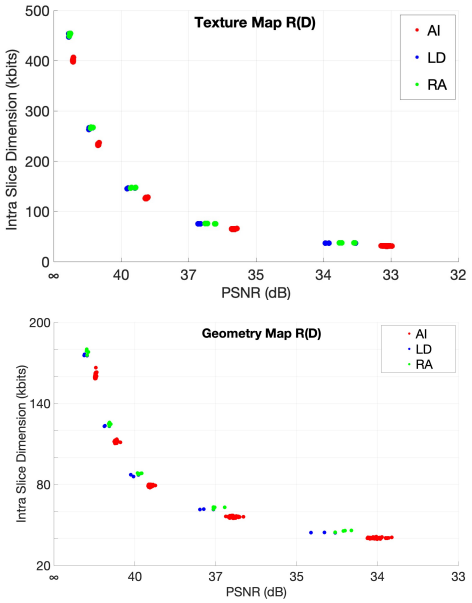

Fig. 10. Example of Rate (Intra slice size) versus Distortion (decreasing PSNR) curve for the Texture and Geometry Maps.

For completeness, we report in Fig.11 the autocorrelation coefficient ρ_k versus the lag k for the sequence 'Soldier' (Geometry map, encoded at rate Q3 in Tab.I, Random Access configuration).

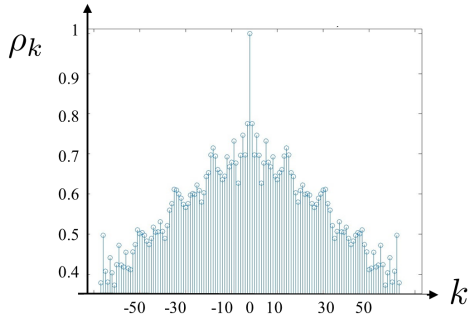

Fig. 11. Autocorrelation coefficient ρ_k versus the lag k (Soldier sequence, Geometry map, rate Q3 in Tab.I, Random Access).

VII. Conclusion and future work

This paper provides an empirical XR traffic model useful for next generation network design purposes. We propose a two-tier model: the first stage addresses coarse-grain traffic fluctuations at different time scales, ranging from several minutes for user behaviour, to seconds for rate adaptation; the second stage relates to fine-grain fluctuations of tenths of milliseconds of the sole encoded XR data packet. The first stage encompasses a hidden (semi) Markov model of the traffic fluctuations while the second stage resort to a conditional heavy tailed data distribution, found by numerical studies on real encoded data. The encoded XR data traces and a Matlab XR source simulator developed during this study have been released. The empirical model is driven by few parameters and it can be adopted as a tool in network design, for the purposes of XR oriented network slicing, orchestration or cost-revenue analysis in next generation networks.

References

[1] Wirth, F., Quehl, J., Ota, J., Stiller, C. (2019, June). Pointatme: efficient 3d point cloud labeling in virtual reality. In 2019 IEEE Intelligent Vehicles Symposium (IV) (pp. 1693-1698). IEEE.

[2] Navarro-Ortiz, J., Romero-Diaz, P., Sendra, S., Ameigeiras, P., Ramos-Munoz, J.J. and Lopez-Soler, J.M., 2020. A survey on 5G usage scenarios and traffic models. IEEE Communications Surveys & Tutorials, 22(2), pp.905-929.

[3] Rossi, L., Chakareski, J., Frossard, P. and Colonnese, S., 2014. A poisson hidden markov model for multiview video traffic. IEEE/ACM transactions on Networking, 23(2), pp.547-558.

[4] Sun, Shuai, Shaoxi Li, Yan Li, Bill Moran, and Wayne ST Rowe. "Smartphone user tracking by incorporating user orientation using a double-layer HMM." IEEE Transactions on Vehicular Technology 71, no. 7 (2022): 7780-7790.

[5] Hosseini, M. and Timmerer, C., 2018, June. Dynamic adaptive point cloud streaming. In Proceedings of the 23rd Packet Video Workshop (pp. 25-30).

[6] Carrascosa, M. and Bellalta, B., 2022. Cloud-gaming: Analysis of google stadia traffic. Computer Communications, 188, pp.99-116.

[7] Cao, C., Preda, M. and Zaharia, T., 2022, October. Affine Transformation-Based Color Compression For Dynamic 3D Point Clouds. In 2022 IEEE International Conference on Image Processing (ICIP) (pp. 1556-1560). IEEE.

[8] Aceto, G., Bovenzi, G., Ciuonzo, D., Montieri, A., Persico, V. and Pescapé, A., 2021. Characterization and prediction of mobile-app traffic using Markov modeling. IEEE Transactions on Network and Service Management, 18(1), pp.907-925.

[9] Guarino, I., Nascita, A., Aceto, G. and Pescapà, A., 2021, September. Mobile network traffic prediction using high order Markov chains trained at multiple granularity. In 2021 IEEE 6th International Forum on Research and Technology for Society and Industry (RTSI) (pp. 394-399). IEEE.

[10] Morin, D.G., Medda, D., Iossifides, A., Chatzimisios, P., Armada, A.G., Villegas, A. and Perez, P., 2023. An eXtended Reality Offloading IP Traffic Dataset and Models. arXiv preprint arXiv:2301.11217.

[11] Lecci, M., Drago, M., Zanella, A. and Zorzi, M., 2021. An open framework for analyzing and modeling XR network traffic. IEEE Access, 9, pp.129782-129795.

[12] Nguyen, Minh, Shivi Vats, Sam Van Damme, Jeroen Van Der Hooft, Maria Torres Vega, Tim Wauters, Christian Timmerer, and Hermann Hellwagner. "Impact of Quality and Distance on the Perception of Point Clouds in Mixed Reality." In 2023 15th International Conference on Quality of Multimedia Experience (QoMEX), pp. 87-90. IEEE, 2023.

[13] Xiang, X., Abdein, R., Li, W. and El Saddik, A., 2023. Deep Scene Flow Learning: From 2D Images to 3D Point Clouds. IEEE Transactions on Pattern Analysis and Machine Intelligence.

[14] Woodward, J. and Ruiz, J., 2022. Analytic review of using augmented reality for situational awareness. IEEE Transactions on Visualization and Computer Graphics, 29(4), pp.2166-2183.

[15] Graziosi, D., Nakagami, O., Kuma, S., Zaghetto, A., Suzuki, T. and Tabatabai, A., 2020. An overview of ongoing point cloud compression standardization activities: Video-based (V-PCC) and geometry-based (G-PCC). APSIPA Transactions on Signal and Information Processing, 9, p.e13.

[16] Quach, M., Pang, J., Tian, D., Valenzise, G. and Dufaux, F., 2022. Survey on deep learning-based point cloud compression. Frontiers in Signal Processing.

[17] 3GPP, Technical Specification Group Services and System. Technical Report TR 26.928 V1.2.1, Extended Reality (XR) in 5G. 2020.

[18] Liu, Z., Li, Q., Chen, X., Wu, C., Ishihara, S., Li, J. and Ji, Y., 2021. Point cloud video streaming: Challenges and solutions. IEEE Network, 35(5), pp.202-209

[19] Colonnese, S., Frossard, P., Rinauro, S., Rossi, L. and Scarano, G., 2013. Joint source and sending rate modeling in adaptive video streaming. Signal Processing: Image Communication, 28(5), pp.403-416.

[20] Kua, J., Armitage, G. and Branch, P., 2017. A survey of rate adaptation techniques for dynamic adaptive streaming over HTTP. IEEE Communications Surveys & Tutorials, 19(3), pp.1842-1866.

[21] Akhshabi, S., Begen, A.C. and Dovrolis, C., 2011, February. An experimental evaluation of rate-adaptation algorithms in adaptive streaming over HTTP. In Proceedings of the second annual ACM conference on Multimedia systems (pp. 157-168).

[22] González, S., Castellanos, W., Guzmán, P., Arce, P. and Guerri, J.C., 2016. Simulation and experimental testbed for adaptive video streaming in ad hoc networks. Ad Hoc Networks, 52, pp.89-105.

[23] Jiang, X., Yu, F.R., Song, T. and Leung, V.C., 2021. Resource allocation of video streaming over vehicular networks: a survey, some research issues and challenges. IEEE Transactions on Intelligent Transportation Systems, 23(7), pp.5955-5975.

[24] Maja Krivokuća, Philip A. Chou, and Patrick Savill, "8i Voxelized Surface Light Field (8iVSLF) Dataset," ISO/IEC JTC1/SC29 WG11 (MPEG) input document m42914, Ljubljana, July 2018.

[25] Graziosi, D., O. Nakagami, S. Kuma, A. Zaghetto, T. Suzuki, and A. Tabatabai. "An overview of ongoing point cloud compression standardization activities: Video-based (V-PCC) and geometry-based (G-PCC)." APSIPA Transactions on Signal and Information Processing 9 (2020): e13.

[26] "Video codec based point cloud compression (V-PCC) test model" *https://github.com/MPEGGroup/mpeg-pcc-tmc2* (last accessed June 2023).

[27] Schmidt, S., Zadtootaghaj, S., Sabet, S.S. and Möller, S., 2021, June. Modeling and understanding the quality of experience of online mobile gaming services. In 2021 13th International Conference on Quality of Multimedia Experience (QoMEX) (pp. 157-162). IEEE.

[28] *https://github.com/TizianaCattai*, last accessed MAy 2024.

[29] Nguyen, M., Vats, S., Van Damme, S., Van der Hooft, J., Vega, M. T., Wauters, T., ... Hellwagner, H. (2023). Characterization of the Quality of Experience and Immersion of Point Cloud Video Sequences through a Subjective Study. Ieee Access.