A Distributed Approach for Virtual Reality Application in Cellular Networks

Tri Nguyen Dang, Choong Seon Hong

Department of Computer Science and Engineering
Kyung Hee University, 446-701, Korea
email: {trind, cshong}@khu.ac.kr

Abstract

The fifth generation of the cellular network (5G) has been introduced in recent years. It has brought out various types of applications required ultra-low latency (URLLC) and high bandwidth, i.e, Augmented Reality (AR), Virtual Reality (VR). These kinds of applications consumed a large number of resources, i.e, communication, computation resource, etc. Thus, the Multi-access Edge Computing server may run-out of resources due to the high demand of users. Therefore, caching is one of the solutions to save computational or communication resources. In this paper, we propose a framework to optimize caching strategy for AR content at the user equipment, i.e, AR gear, smartphone. The optimization problem is the combinatorial category, thus, we propose an equivalent problem based on the Alternating Direction Method of Multipliers (ADMM). We then, validate our proposed method with numerical results with a different setting in Julia language.

1 Introduction

The remarkable prevalence of 5G has enabled various types of VR/AR applications for end-user devices. In which, VR application required URLLC, and high bandwidth i.e, less than $20ms$ delay, $6GBs/s$ bandwidth [1]. However, due to the limitation of the physical resource of current technology, the MEC cannot serve all demand from the end-user (UE). The more number of demand leads to a high probability of the problem of network congestion or MEC overload. Thus, the quality-of-service (QoS) might lose, and demand of UE can not be served. Several works in [2], [3], [4], [5] propose methods of proactive caching to reduce computational resources utilization as well as service delay. However, the most popular of AR application is 360^0 video content, and, has a very limited study on optimizing streaming performance. The different between 360^0 video content and regular video is lead to many challenging. For example, each chunk of regular video only has 120^0 scene but 360^0 video including at least 5 scenes for each chunk i.e, main scene, left scene, right scene, down scene, and up scene (illustrate in Fig. 1). In order to optimize the performance of streaming 360^0 video, we have to decide which part of the chunk will be served in the high quality, and which chunk will be served in lower quality. Thus, we can save bandwidth by reducing the total number of content size on transferring from BS to UE.

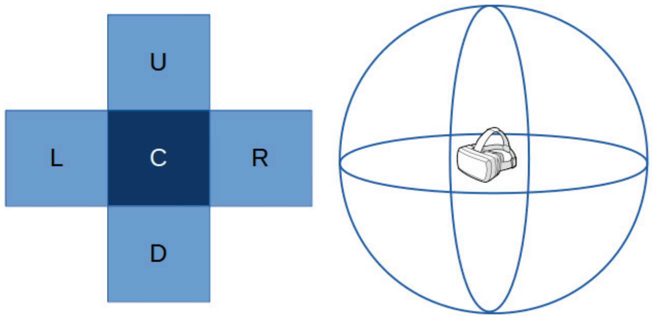

Fig. 1: Illustration of our system model.

2 System Model

In this paper, we consider a network area consisting a single base station equipped MEC capability; a 360^0 video content $\mathbf{V}$, with a sequence of chunks for each time slot denoted as $\mathbf{V} = \{V(1), V(2), \ldots, V(t), \ldots, V(T)\}$, where T (seconds) is the length of $\mathbf{V}$. Each time slot $t \in T$, $s(t) = \{s_c(t), s_l(t), s_r(t), s_u(t), s_d(t)\}$ is a positive tuple represent for the size of chunks $V(t)$, i.e, $s_l(t)$ is the left side, $s_r(t)$ is the left side, $s_c(t)$ is the central, $s_u(t)$ is the upside, and, $s_d(t)$ is the downside (illustrate in Fig. 1). In 360^0 video content, UE cannot watch all scenes at once. Each time slot, UE can watch at most one scene because of human eye ability only can view the maximum 120^0. Therefore, at a single time slot t, only the central scene will display with the highest quality, and the remaining sides are display or load with lower quality or blur. Thus, we can save the bandwidth to transfer content

$V(t)$ from the MEC to the UE. Moreover, based on this property the MEC can make a decision in which chunk will be served at a higher quality than the other chunks. The more number of chunks is displayed at the lower quality the more bandwidth can be saved. However, the Quality-of-Experience (QoE) may reduce because of the wrong decision made by the BS. It means the scene UE watching is served at a lower quality than expected. Thus, the MEC needs to optimization scheme in order to improve the QoE, and, reduce total data traffic in the network. In the next section, we present our proposed method and solution approach.

3 Problem Formulation

Let $\mathbf{p}(t) = \{p_c(t), p_l(t), p_r(t), p_u(t), p_d(t)\}$ be the popularity of scenes at time slot t. $\mathbf{p}(t)$ can be modeled follow the Zipf distribution with parameter $\alpha(t)$,

$$p_k = \frac{1/k^{\alpha(t)}}{\sum_{m=1}^{M} 1/m^{\alpha(t)}}. \quad (1)$$

Furthermore, we assume that the transition probability from time slot t to $t+1$ is independent identically distribution (i.i.d). On the other hand, in the content centric network (CCN), most of the utility functions are measured via mean-opinion-score (MOS). MOS is measuring the satisfaction of end-users via direct asking about UE satisfy with video perception. However, this approach is costly and hard to obtain in reality. Therefore, we approximate the utility function under the linear scale of the logarithm function Hence, the expected utility of chunk $V(t)$ is calculated as follows:

$$\mathbf{E}(t) = \mathbf{p}^T(t)\mathbf{u}(t), \forall t \in T. \quad (2)$$

where $\mathbf{u}(t) = [\log(s_c(t)), \ldots, \log(s_d(t))]$. Let $\mathbf{x}(t) = \{x_c(t), x_l(t), x_r(t), x_u(t), x_d(t)\}$ be decision variables of the BS according to each side of chunk, respectively. We then formulate the optimization problem as follows:

$$\max_{\mathbf{x}} \sum_{t=1}^{T} \left[\mathbf{u}(t) \circ \mathbf{p}(t)\right]^T \mathbf{x}(t), \quad (3a)$$

$$\text{s.t.} \quad (3b)$$

$$\text{C1}: \mathbf{s}(t)^T \mathbf{x}(t) \leq S, \forall t \in T, \quad (3c)$$

$$\text{C2}: \mathbf{1}^T \mathbf{s}(t)/r_{max}(t) \leq l_{max}, \forall t \in T, \quad (3d)$$

$$\mathbf{x}(t) \in \mathbb{B}^5, \mathbf{x} \in \mathbb{B}^{5\times T}.$$

where $\circ$ in (3a) is the element-wise product of two vectors $\mathbf{u}$, and, $\mathbf{p}$. The objective is maximize total expected utility of the system. Subject to the constraint (C1) represent for the storage capacity of UE. It means that each time slot only some of the chunk can be buffered at user equipment and not exceed the total capacity. Constraint (C2) is the URLLC requirement. In which, the transmission delay is less then or equal to maximum acceptable delay $l_{max} = 20$(ms). Due to binary variables $\mathbf{x}(t)$ the problem in (3) fall into combinatorial category, thus, obtain an optimal solution is intractable. We then propose an equivalent problem by relaxing the binary variable in to continuous variables and apply the ADMM method [6] to achieve an optimal solution via iterative algorithm. Firstly, we introduce a new auxiliary variable $z(t)$ such that $\mathbf{x}(t) = z(t)$. Then rewrite an equivalent problem as follows:

$$\max_{\mathbf{x}} \sum_{t=1}^{T} f(\mathbf{x}(t)) + g(z(t)) \quad (4a)$$

$$\text{s.t.} \quad (4b)$$

$$\text{C1}: \mathbf{s}(t)^T \mathbf{x}(t) \leq S, \forall t \in T, \quad (4c)$$

$$\text{C2}: \mathbf{1}^T \mathbf{s}(t)^T/r_{max}(t) \leq l_{max}, \forall t \in T, \quad (4d)$$

$$\text{C3}: \mathbf{x}(t) = z(t), \forall t \in T, \quad (4e)$$

$$\text{C4}: \mathbf{1}^T \mathbf{x}(t) = 1, \forall t \in T, \quad (4f)$$

$$\mathbf{x}(t) \in R^5, \mathbf{x} \in \mathbf{R}^{5\times T}, 0 \leq \mathbf{x}(t), z(t) \leq 1.$$

where $\mathbf{x}(t) = \{x_l(t), x_r(t), x_u(t), x_d(t), x_c(t)\}, \forall t \in T$. We then re-write feasible set of (4):

$$\mathcal{D} = \{x | x \in \mathbb{R}^5, (C1), (C2), (C4), 0 \leq x \leq 1\}. \quad (5)$$

And, $g(z(t))$ is a projection function of $z(t)$ onto set $\mathbf{D}$, where

$$g(z(t)) = \Pi_{\mathbf{D}}(z(t)) = \begin{cases} 0, & \text{if } z(t) \in \mathcal{D}, \\ -\infty, & \text{otherwise.} \end{cases} \quad (6)$$

Based on (4), (5), and (6), the ADMM-equivalent problem can be formulated as:

$$\max_{\mathbf{x}} \sum_{t=1}^{T} f(\mathbf{x}(t)) + g(z(t)) \quad (7a)$$

$$\text{s.t.} \quad (7b)$$

$$\mathbf{x}(t) = z(t), \forall t \in T, \quad (7c)$$

$$x \in \mathcal{D}.$$

Following the ADMM method, the Lagrangian function can be formulated as follows:

$$\mathcal{L}_\rho(\mathbf{x}, z, \lambda) = \sum_{t=1}^{T} f(\mathbf{x}(t)) + g(z(t)) + \lambda^T \left(\mathbf{x}(t) - z(t)\right) + \frac{\rho}{2}||\mathbf{x}(t) - z(t)||_2^2. \quad (8)$$

Algorithm 1 ADMM-based Algorithm

Input: $\mathbf{V}, T, \mathbf{p}, \mathbf{s}, K$;
Output: Maximum expected utility;
Initialize $\mathbf{x}(t)^0, \mathbf{z}(t)^0, \lambda^0, \rho \leftarrow 0.5$;
for $k \in K$ **do**
  Update sequence variables $\mathbf{x}, \mathbf{z}, \lambda$ in (9), (10), (11), respectively;
  Update the objective function in (4);
end for

Based on (8), the sequence update of primal variable $\mathbf{x}$, the auxiliary variable $\mathbf{z}$, and the Lagrangian multiplier λ can be formulated as follows:

$$\mathbf{x}(t)^{k+1} = \arg\min\{-f(x(t)) + \lambda^{k,T}\left(\mathbf{x}(t) - \mathbf{z}^k(t)\right) + \frac{\rho}{2}||\mathbf{x}(t) - \mathbf{z}^k(t)||_2^2\}. \tag{9}$$

$$\mathbf{z}(t)^{k+1} = \arg\min\{-g(z(t)) + \lambda^{T}\left(\mathbf{x}^{k+1}(t) - \mathbf{z}(t)\right) + \frac{\rho}{2}||\mathbf{x}^{k+1}(t) - \mathbf{z}(t)||_2^2\}. \tag{10}$$

$$\lambda^{k+1} = \lambda^k - \rho(\mathbf{x}^{k+1}(t) - z^{k+1}(t)). \tag{11}$$

We then propose an iterative algorithm describe in Alg. 1 to solve the problem in (4).

4 Numerical Results

In this paper, we use Julia language, and Convex.jl [7] as our simulation tools. For the simulation setup, the length of AR content $T = 3600$ (seconds). The size of is chunk is randomly in range $[1.0, 5.0]$ (GBs), the maximum bandwidth $r_{max}(t)$ is 3(GBs/s), the Zipf parameter α is randomly in range $[0.8, 1.5]$, and the maximum iteration $K = 100$. As shown

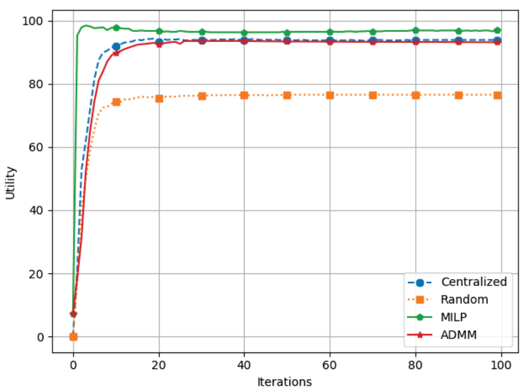

Fig. 2: Numerical results.

in Fig. 2, our scheme achieve optimal solution with the same performance of centralized scheme and fast convergence after 10 iterations, and close to the global optimal solution via Mixed-integer Linear Programming (MILP) method.

5 Conclusion

In this paper, we proposed a distributed approach for VR content streaming in 5G network. The optimization problem is an combinatorial family, we then apply the ADMM method to obtain an optimal solution with the same performance in centralized scheme. By applying this scheme, we can saved the total data traffic in the network and avoid the network congestion problem.

Acknowledgments

This research was supported by the MSIT (Ministry of Science and ICT), Korea, under the Grand Information Technology Research Center support program (IITP-2020-2015-0-00742) supervised by the IITP (Institute for Information & communications Technology Planning Evaluation). Dr. CS Hong is the corresponding author.

References

[1] W. Saad, M. Bennis, and M. Chen, "A vision of 6g wireless systems: Applications, trends, technologies, and open research problems," *IEEE Network*, pp. 1–9, 2019.

[2] A. Ndikumana, S. Ullah, and C. S. H. Do Hyeon Kim, "Deepauc: Joint deep learning and auction for congestion-aware caching in named data networking," *PloS one*, vol. 14, no. 8, 2019.

[3] K. Thar, S. Ullah, R. Haw, T. LeAnh, T. Z. Oo, and C. S. Hong, "Hybrid caching and requests forwarding in information centric networking," in *2015 17th Asia-Pacific Network Operations and Management Symposium (APNOMS)*. IEEE, 2015, pp. 203–208.

[4] S. Ullah, L. U. Khan, and C. S. Hong, "Socially-aware svc video caching in information centric networking with d2d communication," *Journal of Korean Information Science Society*, pp. 1389–1391, 2018.

[5] K. Thar, N. H. Tran, S. Ullah, T. Z. Oo, and C. S. Hong, "Online caching and cooperative forwarding in information centric networking," *IEEE Access*, vol. 6, pp. 59 679–59 694, 2018.

[6] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein *et al.*, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine learning*, vol. 3, no. 1, pp. 1–122, 2011.

[7] M. Udell, K. Mohan, D. Zeng, J. Hong, S. Diamond, and S. Boyd, "Convex optimization in Julia," in *Proceedings of the 1st First Workshop for High Performance Technical Computing in Dynamic Languages*. IEEE Press, 2014, pp. 18–28.