Joint Scheduling of Communication-Computation-Caching in F-RAN

Zishuo You*, Qiang Li*, Ashish Pandharipande†, Xiaohu Ge*
*Huazhong University of Science and Technology, 430074 P. R. China,
†NXP Semiconductors, 5656 AE Eindhoven, Netherlands

***Abstract*—With the emergence of new services and applications, it becomes indispensable to jointly allocate heterogeneous network resources in face of various quality-of-experience (QoE) requirements. In order to quantitatively characterize the tradeoff between communication, computation and caching (3C), which is still unclear, a typical application of adaptive bitrate (ABR) content streaming is considered in fog radio access networks (F-RAN). For improving the QoE, an optimization problem of minimizing the average delay is first formulated subject to constrained 3C resources. To solve the resulting problem that is NP-hard, it is first relaxed as a linear programming problem, then an alternating direction method of multipliers (ADMM)-based algorithm is proposed, which has the advantages of relatively low complexity and fast convergence. Simulation results show that with a joint allocation of 3C, significant performance gains are achieved by the proposed ADMM-based algorithm. Furthermore, while the scarcity of one resource can be compensated by other resources to a certain extent, it requires a matched allocation of 3C resources to effectively improve QoE and at the same time avoid resource waste.**

***Index Terms*—Communication-computation-caching, fog computing, adaptive bitrate streaming, QoE, ADMM.**

I. Introduction

According to the prediction of CISCO, the number of devices connected to networks will increase from 18.4 billions in 2018 to 29.3 billions in 2023, with the explosive growth of data volume [1]. Due to heterogeneous devices and dynamic network conditions, it is required to provide differentiated and customized quality-of-experience (QoE) to users under different business scenarios. To meet the diverse QoE requirements with efficient utilization of the limited network resources, fog computing has been proposed, which pushes major network functions to the edge by scheduling communication, computation and caching (3C) resources in the radio access networks [2], [3].

Although it is widely recognized that 3C are mutually constrained and complementary, the quantitative tradeoff between 3C is still unclear. Despite the advantages inherent in the joint scheduling of 3C, a mismatch in the allocation of 3C resources may result in an unnecessary performance bottleneck and resource waste [4]. In order to better support the users' diverse QoE requirements, how to break the network performance bottleneck through the coordination and complementation of 3C scheduling has become an urgent problem to be solved [5].

With the joint design of communication and caching, some popular contents can be cached at fog nodes and obtained directly from the edge [6]-[8], which is able to save the bandwidth and reduce the delay by exploiting the tradeoff between communication and caching. With the joint design of communication and computation, computation-intensive tasks can be offloaded to fog nodes or cloud [9]-[11], which is able to reduce energy consumption and delay by exploiting the tradeoff between communication and computation. With the joint design of computation and caching, the adaptive bitrate (ABR) streaming was considered [12]-[15], where cached contents of higher bitrate versions can be transcoded to lower bitrate versions to satisfy the diverse QoE requirements, by exploiting the tradeoff between computation and caching.

In addition, extensive efforts on the tradeoff of 3C have been made. The joint computing, caching, communication, and control at the edge was investigated to minimize a linear combination of the bandwidth consumption and network latency in [16]. Since the formulated problem was non-convex, a proximal upper bound problem of it was proposed, which was solved by the block successive upper bound minimization method. A joint 3C scheduling model for smart cars was considered in [17], where the caching and transcoding decisions were designed to minimize the content retrieval delay. This NP-hard problem was first relaxed as a linear problem, toward which an alternating direction method of multipliers (ADMM)-based algorithm was then proposed. The joint optimization of content caching, computation offloading and radio resource allocation for fog-enabled Internet of Things was formulated with the goal of minimizing the average end-to-end latency in [18]. The actor-critic deep reinforcement learning algorithm was proposed and results demonstrated the learning capacity and the performance of the proposed algorithm.

For tractable analysis, most aforementioned studies considered a simplistic scenario [16]-[18], where the arrivals of requests were omitted and the processing of user requests was assumed to complete instantaneously. Although collaborative caching among multiple BSs was modeled by a multi-class processor queuing problem in [19], it took no account of the available computing resource.

Motivated by the aforementioned studies, in order to meet the diverse QoE requirements subject to various network

The authors would like to acknowledge the support from the Natural Science Foundation of China (NSFC) under grant 61971461, the Hubei Provincial Key R&D Program under grants 2021BAA015 and 2020BAA002.

GLOBECOM 2022 - 2022 IEEE Global Communications Conference | 978-1-6654-3540-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/GLOBECOM48099.2022.10001674

resource constraints, in this paper we investigate the joint scheduling of 3C in fog radio access networks (F-RANs). The main contents and contributions are summarized as follows.

- To facilitate ABR content streaming applications, a joint 3C scheduling mechanism is proposed in F-RAN. Upon receiving a user request at the fog node, if the requested content with specified bitrate version has been cached, then it will be directly provisioned from the cache; if only a higher bitrate version of the requested content has been cached, then it will be provisioned after transcoding and processing; otherwise, the requested content has to be fetched from the cloud data center (CDC) [12]-[15]. Within this hierarchical framework, services can be promptly provisioned while efficiently utilizing the available resources.
- By modeling the content provision process as a multi-class processor queueing process [19], the joint scheduling of 3C resources is formulated as a problem on minimizing the average delay of content delivery. In order to solve the resulting integer problem that is NP-hard in general, we first relax it as a linear problem, and then propose an ADMM-based caching algorithm. The proposed algorithm is capable of solving distributed convex optimization problems, and has fast processing speed and good convergence performance [20].
- Simulation results show that the proposed algorithm can converge to the global optimal solution with a less number of iterations as compared to the existing algorithms. Furthermore, while the scarcity of one resource can be compensated by other resources to a certain extent, the excessive allocation of one resource will also cause unnecessary waste. Thus, a matched allocation of the heterogeneous network resources of 3C can effectively improve QoE and at the same time efficiently utilize the available resources.

The rest of this paper is organized as follows. The system model is presented in Section II. In Section III, the optimization problem on minimizing the average delay of user requests is formulated and an ADMM-based caching algorithm is proposed. Simulations are carried out in Section IV. Finally, the paper is concluded in Section V.

II. System Model

A. 3C-enabled F-RAN

As shown in Fig. 1, an F-RAN model enabled with the joint 3C scheduling is constructed, where each fog access point (F-AP) is attached with a fog node to provide services to fog user equipments (F-UEs) within its coverage.

To meet the diverse QoE requirements in content streaming, a typical scenario of ABR is considered [21], where the requested contents of different bitrates are provisioned to users depending on the capability of F-UEs, channel conditions and available resources. To be specific, three service provision modes are involved upon receiving a user request [12]-[15]. If the required bitrate version of the content has been cached at the fog node, it will be directly provisioned to the requesting user. If only a higher bitrate version of the requested content has been cached, it will be first transcoded to the requested version and then provisioned to the requesting user. Otherwise, the requested content has to be fetched from the CDC.

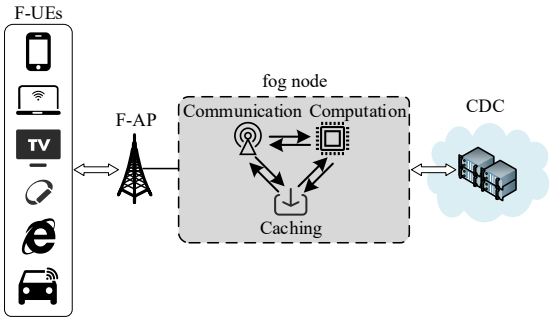

Fig. 1: An illustration of F-RAN model with the joint scheduling of 3C.

Without loss of generality, it is assumed that users request content files from a library consisting of M distinct contents, denoted as $\mathcal{M} = \{1, \ldots, M\}$ [8]. Each content has K possible bitrate versions ranging from the lowest bitrate version $k = 1$ to the highest bitrate version $k = K$, denoted as $\mathcal{K} = \{1, \ldots, K\}$ [22]. The probability that the k-th version of the m-th content is requested, i.e., the popularity of content (m, k), is defined as $P_r(m, k)$. While it is commonly assumed that the content popularity follows a Zipf distribution with parameter α [23], the version of each content is randomly requested following a Uniform distribution [24]. Then for an arbitrary user request, the probability that the content m with version k is requested is given by

$$P_r(m,k) = \frac{m^{-\alpha}}{K \sum_{i=1}^{M} i^{-\alpha}}. \tag{1}$$

For ease of exposition, we define a caching matrix $\mathbf{P} \in \mathbb{R}^{M \times K}$, which indicates whether the content (m, k) is cached or not at the fog node [7]. In order to quantitatively characterize the tradeoff between 3C, we denote the caching resources and computation resources available at the fog node as C and Q [17]. Since the allocated communication resources, e.g., spectrum, are proportional to the data transmission rate [17], [18], without loss of generality, we let R_b and R_e denote the achievable data rates for the downlink transmissions from CDC to F-AP and from F-AP to F-UEs, respectively.

In order to improve users' QoE, next we attempt to investigate the traffic modeling and analyze the average delay of service provision subject to constrained 3C resources.

B. Traffic Modeling and Service Provision Modes

Without loss of generality, it is assumed that the users' request arrivals at the fog node follow a Poisson process with parameter λ [19]. When a user request is generated, it first establishes a connection with the nearest F-AP, through which the requested content can then be directly provisioned upon a cache hit, or can be provisioned after transcoding, or

can be provisioned from the remote CDC upon a cache miss [12]-[15]. Then the cache hit rate H_e, defined as the average probability that the requested content is cached at the fog node, is given by

$$H_e = \sum\nolimits_{m=1}^{M} \sum\nolimits_{k=1}^{K} P_r(m,k) P(m,k). \tag{2}$$

Similarly, the transcoding hit rate H_c, defined as the average probability that a higher bitrate version of the requested content is cached at the fog node, is expressed as

$$H_c = \sum\nolimits_{m=1}^{M} \sum\nolimits_{k=1}^{K-1} P_r(m,k) \sum\nolimits_{k'=k+1}^{K} P(m,k'). \tag{3}$$

When neither the requested content nor a higher bitrate version of the requested content is cached, the corresponding cache miss rate H_b can be expressed as

$$\begin{aligned} H_b &= 1 - \sum_{m=1}^{M} \sum_{k=1}^{K} P_r(m,k) \sum_{k'=k}^{K} P(m,k') \\ &= 1 - H_e - H_c. \end{aligned} \tag{4}$$

By deploying multi-class processor queues, each of the above content provision modes can be modeled as an M/M/1 queuing model at the fog node [19], [25]. Specifically, the user requests can be divided into three sub-queues according to the cache status of the requested content, including the direct provision queue, the transcoding provision queue, and the cloud provision queue. The respective request arrival rates are given by [19]

$$\lambda_e = \lambda H_e, \quad \lambda_c = \lambda H_c, \quad \lambda_b = \lambda H_b. \tag{5}$$

For the direct provision queue, the requested content will be directly delivered to the requesting user, and the corresponding service rate is expressed as [19]

$$\mu_e = \frac{R_e}{s(m,k)}, \tag{6}$$

where $s(m,k)$ is the size of the content.

For the transcoding provision queue, the higher bitrate version will be first transcoded to the requested version and then delivered to the requesting user, with the corresponding service rate given by [19]

$$\mu_c = \frac{1}{\frac{s(m,k)}{R_e} + \frac{s(m,k)z}{Q}}, \tag{7}$$

where z represents the computation workload in terms of CPU cycles per bit required for transcoding cached contents, and $\frac{s(m,k)z}{Q}$ represents the time delay for transcoding the content m [17].

For the cloud provision queue, the requested content will be fetched from the CDC, with the corresponding service rate [19]

$$\mu_b = \frac{1}{\frac{s(m,k)}{R_e} + \frac{s(m,k)}{R_b}}. \tag{8}$$

To ensure the stability of the queue, we have $\mu_e > \lambda_e$, $\mu_c > \lambda_c$ and $\mu_b > \lambda_b$ [19].

The average service delay of each sub-queue includes the average waiting delay and the average processing delay [19], [25], [26]. Then for the aforementioned three provision modes, the corresponding average delay can be respectively expressed as [19]

$$T_e = \frac{\lambda_e}{\mu_e(\mu_e - \lambda_e)} + \frac{1}{\mu_e} = \frac{1}{\mu_e - \lambda_e}, \tag{9a}$$

$$T_c = \frac{\lambda_c}{\mu_c(\mu_c - \lambda_c)} + \frac{1}{\mu_c} = \frac{1}{\mu_c - \lambda_c}, \tag{9b}$$

$$T_b = \frac{\lambda_b}{\mu_b(\mu_b - \lambda_b)} + \frac{1}{\mu_b} = \frac{1}{\mu_b - \lambda_b}. \tag{9c}$$

Thus, for an arbitrary request arriving at the F-AP, the corresponding average delay can be expressed as

$$\begin{aligned} D &= H_e \cdot T_e + H_c \cdot T_c + H_b \cdot T_b \\ &= \frac{H_e}{\mu_e - \lambda H_e} + \frac{H_c}{\mu_c - \lambda H_c} + \frac{H_b}{\mu_b - \lambda H_b}. \end{aligned} \tag{10}$$

Based on the above modeling and analysis, it requires a sophisticated design on the caching matrix $\mathbf{P}$ in order to minimize D. This corresponds to a joint scheduling of 3C resources subject to constrained C, Q, R_b and R_e.

III. Problem Formulation And Algorithm Design

A. Problem Formulation

Subject to constrained 3C resources, in order to efficiently improve the users' QoE, the optimization problem of minimizing the average delay of content provision is formulated as

$$\min_{\mathbf{P}} D \tag{11a}$$

$s.t.,$

$$\sum_{m=1}^{M} \sum_{k=1}^{K} P(m,k) s(m,k) \le C, \tag{11b}$$

$$\lambda \sum_{m=1}^{M} \sum_{k=1}^{K-1} P_r(m,k) \sum_{k'=k+1}^{K} P(m,k') s(m,k) z \le Q, \tag{11c}$$

$$\sum_{k=1}^{K} P(m,k) \le 1, \forall m \in \mathcal{M}, \tag{11d}$$

$$P(m,k) \in \{0,1\}, \forall m \in \mathcal{M}, k \in \mathcal{K}. \tag{11e}$$

The constraint in (11b) guarantees that the size of the contents cached by the fog node cannot exceed the available caching capacity C. The constraint in (11c) guarantees that the number of CPU cycles required on transcoding processing in a unit time cannot exceed the available computation resource Q. The constraint in (11d) guarantees that at most one bitrate version of a content is cached, which ensures the diversity of the cached contents [15]. Finally, the binary decision variable is given by the constraint (11e).

The optimization problem (11) belongs to the integer problem, which is NP-hard in general [6], [7], [17]. Such problems require exponential time to be solved centrally. In order to solve this kind of integer optimization problem, we adopt the

method of relaxing the integer variable to a continuous variable [17], [27], which corresponds to fractional caching of contents at the fog node, i.e.,

$$P(m,k) \in [0,1], \forall m \in \mathcal{M}, k \in \mathcal{K}. \tag{12}$$

Then the slack problem of (11) can be expressed as

$$\min_{\mathbf{P}} D, \quad s.t. \ \text{(11b)}; \text{(11c)}; \text{(11d)}; \text{(12)}. \tag{13}$$

Theorem 1: Based on the caching matrix $\mathbf{P}$, the caching vector can be defined as $\mathbf{p} = \langle P(m,k) \rangle_{m \in \mathcal{M}, k \in \mathcal{K}}$. Then the slack problem (13) corresponds to the convex optimization of D with respect to $\mathbf{p}$.

Proof 1: See Appendix A.

B. Proposed ADMM-Based Caching Algorithm

To solve the slack problem (13) that is convex with respect to $\mathbf{p}$, we propose an ADMM-based caching algorithm, which can quickly converge to the optimal solution with low complexity. However, basic ADMM cannot be directly applied to (13), so we need to transform its constraints [19].

We define a matrix $\mathbf{A} \in \mathbb{R}^{1 \times MK}$, whose elements satisfy $a_{1,j} = s(m,k)$, where $j = m + M(k-1), m \in \mathcal{M}, k \in \mathcal{K}$. Through the defined matrix $\mathbf{A}$, constraint (11b) can be transformed into

$$\mathbf{A}\mathbf{p} \leq C. \tag{14}$$

Then we define a series of matrices $\mathbf{B}_i \in \mathbb{R}^{1 \times MK}$, where $i = 1, \ldots, K-1$, and for matrix $\mathbf{B}_i$, its elements satisfy $b^i_{1,j} = \lambda P_r(m,i) s(m,i) z$, where $j = m + M(k-1)$, $m \in \mathcal{M}$, $k = i+1, \ldots, K$. Otherwise, its elements are $b^i_{1,j} = 0$. Through the defined matrices, constraint (11c) can be transformed into

$$\sum_{i=1}^{K-1} \mathbf{B}_i \mathbf{p} \leq Q. \tag{15}$$

Similarly, we define a matrix $\mathbf{E} \in \mathbb{R}^{M \times MK}$, whose elements satisfy $e_{i,j} = 1$, where $j = i + M(k-1), k \in \mathcal{K}$. Otherwise, its elements are $e_{i,j} = 0$. Through the defined matrix $\mathbf{E}$, constraint (11d) can be transformed into

$$\mathbf{E}\mathbf{p} \leq \mathbf{F}, \tag{16}$$

where $\mathbf{F} = [1, \ldots, 1]^T$.

Based on the transformations (14), (15), (16), the optimization problem (13) can be reformulated as

$$\min_{\mathbf{p}} D(\mathbf{p}), \quad s.t. \ \text{(14)}; \text{(15)}; \text{(16)}; \text{(12)}. \tag{17}$$

In order to transform the problem (17) with inequality constraints into the problem with only equality constraints, we introduce the indicator function [19], [20]

$$I(\mathbf{z}) = \begin{cases} 0, & \mathbf{z} \in \varsigma, \\ \infty, & \mathbf{z} \notin \varsigma, \end{cases} \tag{18}$$

where $\varsigma = \{\mathbf{p} : \text{(14)}; \text{(15)}; \text{(16)}; \text{(12)}\}$ is the the feasible set of $\mathbf{p}$ and $\mathbf{z}$ is the auxiliary variable. Then the optimization problem (17) is transformed into

$$\min_{\mathbf{p}} D(\mathbf{p}) + I(\mathbf{z}), \quad s.t. \ \mathbf{p} = \mathbf{z}. \tag{19}$$

The augmented Lagrangian function of problem (19) is expressed as

$$L_\rho(\mathbf{p}, \mathbf{z}, \mathbf{u}) = D(\mathbf{p}) + I(\mathbf{z}) + \mathbf{u}^T(\mathbf{p} - \mathbf{z}) + \frac{\rho}{2} \|\mathbf{p} - \mathbf{z}\|_2^2, \tag{20}$$

where ρ is the augmented Lagrangian factor and $\mathbf{u}$ is the dual variable. Then the solution can be expressed as [20]

$$\mathbf{p}^{k+1} = \arg\min_{\mathbf{p}} \left[D(\mathbf{p}) + \frac{\rho}{2} \left\| \mathbf{p} - \mathbf{z}^k + \mathbf{u}^k \right\|_2^2 \right], \tag{21a}$$

$$\mathbf{z}^{k+1} = \arg\min_{\mathbf{p}} \left[I(\mathbf{z}) + \frac{\rho}{2} \left\| \mathbf{p}^{k+1} - \mathbf{z} + \mathbf{u}^k \right\|_2^2 \right], \tag{21b}$$

$$\mathbf{u}^{k+1} = \mathbf{u}^k + \mathbf{p}^{k+1} - \mathbf{z}^{k+1}. \tag{21c}$$

The details of the proposed ADMM-based caching algorithm are shown in **Algorithm 1**.

Algorithm 1 Proposed ADMM-Based Caching Algorithm

1: Initialization: $max\_iteration$, primal feasibility tolerance ε_{pri}, dual feasibility tolerance ε_{dual}, $\mathbf{p}^0$, $\mathbf{z}^0$ and $\mathbf{u}^0$;
2: **for** $t = 1, 2, \ldots, max\_iteration$ **do**
3: Update the caching strategy vector: $\mathbf{p}^{t+1} = \arg\min_{\mathbf{p}} \left[D(\mathbf{p}) + \frac{\rho}{2} \|\mathbf{p} - \mathbf{z}^t + \mathbf{u}^t\|_2^2 \right]$;
4: Update the auxiliary vector: $\mathbf{z}^{t+1} = \arg\min_{\mathbf{z}} \left[I(\mathbf{z}) + \frac{\rho}{2} \left\| \mathbf{p}^{t+1} - \mathbf{z} + \mathbf{u}^t \right\|_2^2 \right]$;
5: Update the dual variable: $\mathbf{u}^{t+1} = \mathbf{u}^t + \mathbf{p}^{t+1} - \mathbf{z}^{t+1}$;
6: **if** $\left\| \mathbf{p}^{t+1} - \mathbf{z}^{t+1} \right\| < \varepsilon_{pri} \cap \left\| -\rho(\mathbf{z}^{t+1} - \mathbf{z}^t) \right\| < \varepsilon_{dual}$ **then**
7: Return $D(\mathbf{p}^{t+1})$;
8: Break;
end if
9: $t = t + 1$.
10: **end for**

IV. Simulation Results

In this section, extensive simulations are carried out to evaluate the average delay of content provision under the proposed ADMM-based caching algorithm. For ease of illustration, we let the total number of contents $M = 20$, the number of bitrate versions per content $K = 2$, the arrival rate of user requests at the fog node $\lambda = 1$, and the parameter of the Zipf distribution $\alpha = 0.6$. Although the proposed algorithm is applicable to varying content sizes, we let $s(m,k) = 1, \forall m \in \mathcal{M}, k \in \mathcal{K}$ for simplicity, and let the computation workload $z = 1$. Unless otherwise specified, we let the available cache storage $C = 10$, the CPU cycles per unit time $Q = 10$, and the downlink data transmission rates from CDC to F-AP and from F-AP to F-UEs $R_b = 10$ and $R_e = 5$, respectively.

The convergence performance of the proposed ADMM-based caching algorithm is shown in Fig. 2. It is observed that while both the proposed method and the interior point method (IPM) converge to the optimal solution. The proposed method converges much faster with a less number of iterations.

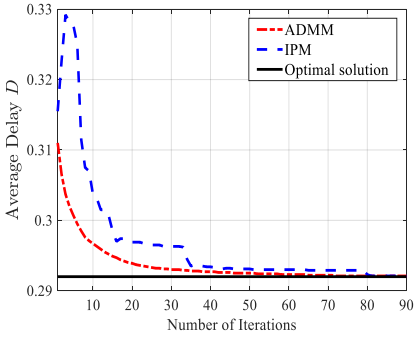

Fig. 2: Comparison of convergence performance.

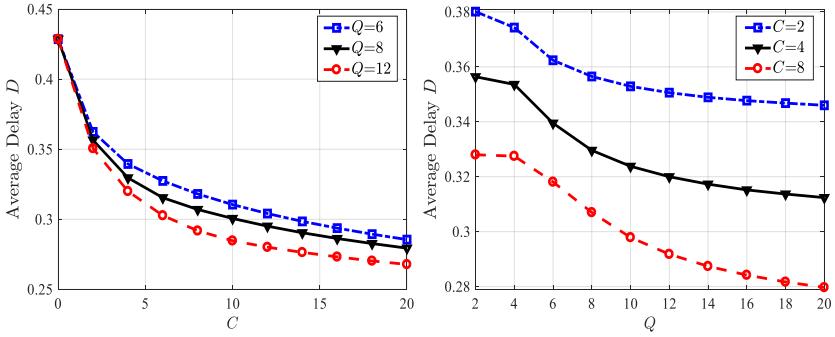

Fig. 3: Average delay D with respect to C and Q.

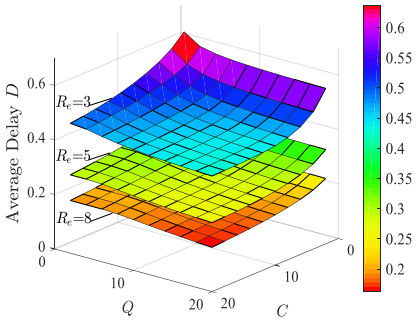

Fig. 4: Average delay D with respect to C and Q under different R_e.

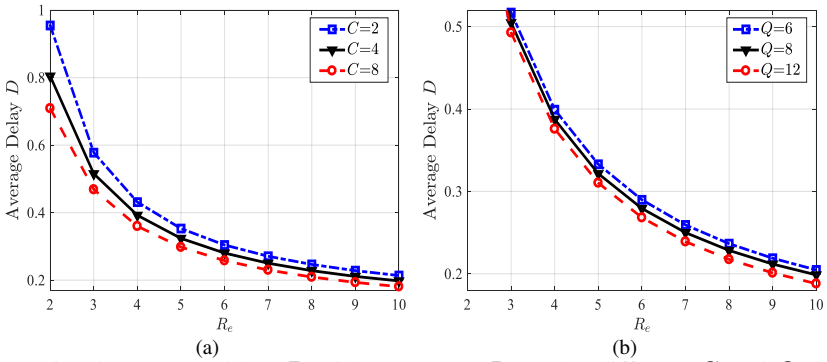

Fig. 5: Average delay D with respect to R_e under different C and Q.

The average delay D achieved by the proposed ADMM-based caching algorithm is shown in Fig. 3 under different values of C and Q. From Fig. 3(a), it is observed that D is improved with increasing C. This is reasonable because more contents can be cached with a larger cache capacity C. This improvement on D becomes slow when C is sufficiently large, where a lower bound exists. This is reasonable because even if all contents can be pre-stored in the cache, the delay performance is still limited by the constrained computation resource Q and the downlink transmission rate R_e. Furthermore, for a given C, it is observed that a better performance can be achieved with a greater Q. This is reasonable as the higher bitrate version of the requested content can be transcoded faster. Similar phenomenon can be observed in Fig. 3(b), where D is reduced with increasing Q and a lower bound, which is determined by the constrained C, R_e and R_b, exists when Q is sufficiently large. For a given Q, additional performance gains can be achieved by adopting a higher C.

To evaluate the effect of the communication resources on the average delay, D is plotted with respect to C and Q in Fig. 4, under different values of R_e. It is observed that while D can be effectively reduced with increasing C and Q, additional performance gains can be achieved with a higher downlink transmission rate R_e. For better illustrations, D is plotted with respect to R_e under different values of C and Q in Fig. 5. It is observed that with a higher R_e, which is able to reduce the downlink transmission delay of the requested content from F-AP to F-UEs, the average delay D can be effectively reduced. Again, for a given R_e, additional performance gains can be achieved by allocating a higher C or Q.

The above observations reveal the quantitative tradeoff between 3C. Since the scarcity of one resource can be compensated by other resources to a certain extent, significant performance gains can be achieved with a joint scheduling of 3C, through which the performance bottleneck due to the scarcity of one resource can be effectively alleviated. On the other hand, it also bears noting that an excessive allocation of one resource will also cause unnecessary resource waste. Thus in a word, generally it requires a matched allocation of 3C resources to guarantee the QoE, while fully utilizing the available resources.

V. CONCLUSIONS

In this paper, a F-RAN model is constructed, based on which a typical application of ABR content streaming is investigated by joint scheduling of 3C. In order to improve the users' QoE, an optimization problem of minimizing the average delay is first formulated. To solve this problem

that is NP-hard in general, we first relax it to a linear programming problem, and then propose an ADMM-based caching algorithm to solve it. Simulation results demonstrate the quantitative tradeoff between 3C. In view of the fact that the scarcity of one resource can be compensated by other resources to a certain extent, the users' QoE can be effectively improved by trading off the heterogeneous network resources with a joint scheduling of 3C. The joint 3C scheduling among multiple fog nodes in a collaborative manner will be delegated to our future work.

APPENDIX A
PROOF OF THEOREM 1

We define the request probability vector $\gamma \in \mathbb{R}^{1\times M}$, denoted as $\gamma = [P_r(1,k),\ldots,P_r(M,k)]$, $\forall k \in \mathcal{K}$, and introduce a series of matrices $\mathbf{A}_0,\ldots,\mathbf{A}_{K-1} \in \mathbb{R}^{M\times MK}$, where

$$a_{i,j}^{0} = \begin{cases} 1, & \text{if } j = i + M(k-1), k = 1,\ldots,K \\ 0, & \text{otherwise} \end{cases}$$
$$\ldots$$
$$a_{i,j}^{K-1} = \begin{cases} 1, & \text{if } j = i + M(k-1), k = K \\ 0, & \text{otherwise} \end{cases} \tag{22}$$

Then the cache hit rate, the transcoding hit rate and the cache miss rate are respectively expressed as $H_e = \gamma\mathbf{A}_0\mathbf{p}$, $H_c = \sum_{k=1}^{K-1}\gamma\mathbf{A}_k\mathbf{p}$, $H_b = 1 - \sum_{k=0}^{K-1}\gamma\mathbf{A}_k\mathbf{p}$.

According to (10), it is easy to prove that the second-order partial derivatives of D with respect to H_e, H_c, H_b are all positive. The Hessian matrix of D can be simplified to $\mathbf{H} = \mathbf{H}_1 + \mathbf{H}_2 + \mathbf{H}_3$, where $\mathbf{H}_1 = \frac{\partial^2 D}{\partial H_e^2}(\gamma\mathbf{A}_0)^T(\gamma\mathbf{A}_0)$, $\mathbf{H}_2 = \frac{\partial^2 D}{\partial H_c^2}\left(\sum_{k=1}^{K-1}\gamma\mathbf{A}_k\right)^T\left(\sum_{k=1}^{K-1}\gamma\mathbf{A}_k\right)$, and $\mathbf{H}_3 = \frac{\partial^2 D}{\partial H_b^2}\left(-\sum_{k=0}^{K-1}\gamma\mathbf{A}_k\right)^{\mathrm{T}}\left(-\sum_{k=0}^{K-1}\gamma\mathbf{A}_k\right)$. We can prove that the determinants of $\mathbf{H}_1$, $\mathbf{H}_2$, $\mathbf{H}_3$ are all zero and the eigenvalues of $\mathbf{H}_1$, $\mathbf{H}_2$, $\mathbf{H}_3$ are non-negative, then $\mathbf{H}$ is a non-negative semi-definite matrix [19]. Thus *Theorem 1* is proved.

REFERENCES

[1] Cisco, et. al., "Cisco annual internet report (2018-2023) white paper," White paper, 2020.

[2] M. Chiang and T. Zhang, "Fog and IoT: an overview of research opportunities," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 854-864, Dec. 2016.

[3] M. Peng, S. Yan, K. Zhang and C. Wang, "Fog-computing-based radio access networks: issues and challenges," *IEEE Netw.*, vol. 30, no. 4, pp. 46-53, July 2016.

[4] X. Wang, W. Cheng and C. Ren, "Multi-objective joint optimization of communication-computation-caching resources in mobile edge computing," *IEEE/CIC ICCC Workshops*, Xiamen, China, 2021, pp. 94-99.

[5] X. Huang, L. He, L. Wang and F. Li, "Towards 5G: joint optimization of video segment caching, transcoding and resource allocation for adaptive video streaming in a multi-access edge computing network," *IEEE Trans. Veh. Technol.*, vol. 70, no. 10, pp. 10909-10924, Oct. 2021.

[7] Q. Li, W. Shi, X. Ge and Z. Niu, "Cooperative edge caching in software-defined hyper-cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 11, pp. 2596-2605, Nov. 2017.

[6] Q. Li, W. Shi, Y. Xiao, X. Ge and A. Pandharipande, "Content size-aware edge caching: a size-weighted popularity-based approach," *IEEE GLOBECOM*, Abu Dhabi, United Arab Emirates, 2018, pp. 206-212.

[8] M. Lei, Q. Li, R. Wu, A. Pandharipande and X. Ge, "Deep deterministic policy gradient-based edge caching: an inherent performance tradeoff," *IEEE GLOBECOM*, Madrid, Spain, 2021, pp. 1-7.

[9] L. Liu, Z. Chang, X. Guo, S. Mao and T. Ristaniemi, "Multiobjective optimization for computation offloading in fog computing," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 283-294, Feb. 2018.

[10] Z. Zhao et al., "On the design of computation offloading in fog radio access networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 7, pp. 7136-7149, July 2019.

[11] X. Gao, X. Huang, S. Bian, Z. Shao and Y. Yang, "PORA: predictive offloading and resource allocation in dynamic fog computing systems," *IEEE Internet Things J.*, vol. 7, no. 1, pp. 72-87, Jan. 2020.

[12] Z. Li, R. Xie, Q. Jia and T. Huang, "Energy-efficient joint caching and transcoding for HTTP adaptive streaming in 5G networks with mobile edge computing," *IEEE ICC Workshops*, Kansas City, USA, 2018, pp. 1-6.

[13] L. Li, D. Shi, R. Hou, R. Chen, B. Lin and M. Pan, "Energy-efficient proactive caching for adaptive video streaming via data-driven optimization," *IEEE Internet Things J.*, vol. 7, no. 6, pp. 5549-5561, June 2020.

[14] Y. Liu, F. R. Yu, X. Li, H. Ji, H. Zhang and V. C. M. Leung, "Joint access and resource management for delay-sensitive transcoding in ultra-dense networks with mobile edge computing," *IEEE ICC*, Kansas City, USA, 2018, pp. 1-6.

[15] Y. Hao, L. Hu, Y. Qian and M. Chen, "Profit maximization for video caching and processing in edge cloud," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 7, pp. 1632-1641, July 2019.

[16] A. Ndikumana et al., "Joint communication, computation, caching, and control in big data multi-access edge computing," *IEEE. Trans. Mob. Comput.*, vol. 19, no. 6, pp. 1359-1374, June 2020.

[17] S. M. A. Kazmi et al., "Infotainment enabled smart cars: a joint communication, caching, and computation approach," *IEEE Trans. Veh. Technol.*, vol. 68, no. 9, pp. 8408-8420, Sep. 2019.

[18] Y. Wei, F. R. Yu, M. Song and Z. Han, "Joint optimization of caching, computing, and radio resources for fog-enabled IoT using natural actor-critic deep reinforcement learning," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2061-2073, April 2019.

[19] Q. Li, Y. Zhang, Y. Li, Y. Xiao and X. Ge, "Capacity-aware edge caching in fog computing networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 9244-9248, Aug. 2020.

[20] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1-122, Jan. 2011.

[21] C. Li, L. Toni, J. Zou, H. Xiong and P. Frossard, "QoE-driven mobile edge caching placement for adaptive video streaming," *IEEE Trans. Multimedia*, vol. 20, no. 4, pp. 965-984, April 2018.

[22] Y. Wang, Y. Zhang, M. Sheng and K. Guo, "On the interaction of video caching and retrieving in multi-server mobile-edge computing systems," *IEEE Wirel. Commun. Lett.*, vol. 8, no. 5, pp. 1444-1447, Oct. 2019.

[23] L. Breslau, P. Cao, L. Fan, G. Phillips and S. Shenker, "Web caching and zipf-like distribution: evidence and implications," *IEEE INFOCOM*, New York, USA, 2002, pp. 126-134.

[24] X. Xu, J. Liu and X. Tao, "Mobile edge computing enhanced adaptive bitrate video delivery with joint cache and radio resource allocation," *IEEE Access*, vol. 5, pp. 16406-16415, Aug. 2017.

[25] L. Klennrock, "Queueing Systems: Theory," vol. 1, New York, USA, 1975.

[26] M. K. Karray and M. Jovanovic, "A queueing theoretic approach to the dimensioning of wireless cellular networks serving variable-bit-rate calls," *IEEE Trans. Veh. Technol.*, vol. 62, no. 6, pp. 2713-2723, Jul. 2013.

[27] Q. Chen, F. R. Yu, T. Huang, R. Xie, J. Liu and Y. Liu, "Joint resource allocation for software-defined networking, caching, and computing," *IEEE-ACM Trans. Netw.*, vol. 26, no. 1, pp. 274-287, Feb. 2018.