

Distributed Edge Caching in Ultra-dense Fog Radio Access Networks: A Mean Field Approach

Yabai Hu, Yanxiang Jiang, *Member, IEEE*, Mehdi Bennis, *Senior Member, IEEE*, and Fu-Chun Zheng, *Senior Member, IEEE*,

Abstract—In this paper, the edge caching problem in ultra-dense fog radio access networks (F-RAN) is investigated. Taking into account time-variant user requests and ultra-dense deployment of fog access points (F-APs), we propose a dynamic distributed edge caching scheme to jointly minimize the request service delay and fronthaul traffic load. Considering the interactive relationship among F-APs, we model the caching optimization problem as a stochastic differential game (SDG) which captures the temporal dynamics of F-AP states and incorporates user requests status. The SDG is further approximated as a mean field game (MFG) by exploiting the ultra-dense property of F-RAN. In the MFG, each F-AP can optimize its caching policy independently through iteratively solving the corresponding partial differential equations without any information exchange with other F-APs. The simulation results show that the proposed edge caching scheme outperforms the baseline schemes under both static and time-variant user requests.

Index Terms—Fog radio access networks, distributed edge caching, mean field game, request service delay, fronthaul traffic load.

I. INTRODUCTION

The increasing number of mobile devices and surging user demand for multi-media service have triggered massive amounts of data in wireless networks, which brings about huge pressure on fronthaul links. Fog radio access networks (F-RAN) can address this problem by allocating caching and computing resources to fog access points (F-APs) deployed ultra-densely at the network edge. In ultra-dense F-RAN, large deployment density enables F-APs to provide large coverage area and high quality wireless links, while edge caching alleviates fronthaul traffic pressure and decreases request service delay by bringing contents closer to users [1]–[3].

Due to the limited caching space of F-APs, the optimization of caching placement is a major issue in F-RAN. The caching placement design can be centralized or distributed. In centralized schemes, caching policies are generally constructed by a control center that collects information of the entire network for seeking a global optimal solution. However, the difficulty

in information acquirement and global optimization renders centralized schemes impractical in ultra-dense networks. In distributed caching schemes, on the contrary, edge caching nodes can independently optimize their caching policies based solely on local information. In [4], a belief propagation (BP)-based distributed caching algorithm was put forward, where each base station computes the caching placement matrix through message passing between nodes. In [5], the cost of fronthaul usage was minimized by using a distributed algorithm based on alternating direction method of multipliers (ADMM). The authors of [6] modeled the network dynamics in a game-theoretic approach and proposed a distributed caching scheme to minimize the backhaul traffic load. A Markov-chain-based optimization framework was established to minimize the macro-cell load by considering high-mobility scenarios in [7].

However, all the above works are based on the assumption of a static content popularity distribution, whereas the user requests for popular contents change over time in practice. Hence, a dynamic distributed edge caching scheme to meet the time-variant demand of users is needed. In [8] and [9], the dynamic evolution of content popularity was investigated using differential equations and Markov chains, respectively. Nevertheless, none of these works account for both request service delay and fronthaul traffic load.

Motivated by the aforementioned discussions, we propose a distributed edge caching scheme adaptive to dynamic user requests in ultra-dense F-RAN. To jointly minimize the request service delay and fronthaul traffic load, a stochastic differential game (SDG) is formulated to model the edge caching problem. The SDG captures the interactions among F-APs and the dynamics of F-AP states, incorporating the status of temporal user requests. Due to the high computational complexity of obtaining the solution, the SDG is approximated as a mean field game (MFG) to solve this challenging problem by utilizing the ultra-dense property of F-RAN. In the MFG, the individual states of all the F-APs can be approximated by a statistical mean field distribution. Then, each F-AP is able to determine the dynamic optimal caching policy based on its local state and the mean field distribution without extra information exchange with the cloud server or adjacent F-APs.

II. SYSTEM MODEL

A. Network Model

Consider an edge caching enabled F-RAN consisting of I F-APs, K users, and one remote cloud server, in which the F-APs

Y. Hu and Y. Jiang are with the National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China, the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China, and the Key Laboratory of Wireless Sensor Network & Communication, Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences, Shanghai 200050, China. (e-mail: yxjiang@seu.edu.cn)

M. Bennis is with the Centre for Wireless Communications, University of Oulu, Oulu 90014, Finland. (e-mail: bennis@ee.oulu.fi)

F. Zheng is with the School of Electronic and Information Engineering, Harbin Institute of Technology, Shenzhen 518055, China, and the National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China. (e-mail: fzheng@ieec.org)

are ultra-densely deployed and connected with the cloud server via fronthaul links. The F-AP set and the user set are denoted by $\mathcal{I} = \{1, 2, \dots, i, \dots, I\}$ and $\mathcal{K} = \{1, 2, \dots, k, \dots, K\}$, respectively. Assume that F-APs are uniformly distributed, users are randomly distributed, and each user associates with a default serving F-AP which is the geographically nearest one. The users with the same default serving F-AP i form the serving user cluster \mathcal{U}^i , and each F-AP serves at most one user at a time. Assume that users request files from the content library $\mathcal{F} = \{f_1, f_2, \dots, f_n, \dots, f_N\}$. Without loss of generality, each file is assumed to have the same size of S bits. The request arrival in consecutive time slots is modeled as a Poisson process, and the t th time slot is denoted as time t for the sake of simplicity. Users receive the requested files from the corresponding F-APs through wireless links, and the transmission rate from F-AP i to its serving user k at time t can be expressed as follows:

$$R_{i,k}(t) = W \log_2 \left(1 + \frac{|h_{i,k}(t)|^2 P_i}{\sigma^2 + \sum_{j \neq i, j \in \mathcal{I}} |h_{j,k}(t)|^2 P_j} \right), \quad (1)$$

where W is the transmission bandwidth, P_i is the transmission power of F-AP i , σ^2 is the noise power, and $h_{i,k}(t)$ is the channel coefficient between F-AP i and user k at time t . By considering a time-varying fading channel, the evolution of $h_{i,k}(t)$ can be characterized by a mean-reverting Ornstein-Uhlenbeck process as follows [10]:

$$dh_{i,k}(t) = \frac{1}{2} \alpha (\mu_h - h_{i,k}(t)) dt + \sigma_h dB_{i,k}(t), \quad (2)$$

where μ_h and σ_h with positive values characterize the long-term mean and variance of the process, respectively, α is the speed of the process towards the long-term mean, and $B_{i,k}(t)$ is a standard Brownian motion that depicts the stochastic dynamics of wireless channels.

B. Content Caching and Delivery Model

Regarding the content processing policy of each F-AP, each request time slot is divided into two phases: caching placement phase and content delivery phase. The storage space of the cloud server is assumed to be considerably large that contains the whole content library, while each F-AP is equipped with limited caching space that can store up to C bits.

In the caching placement phase, F-AP i fetches certain fractions of files through the fronthaul link according to its caching policy which is denoted by the caching vector $\mathbf{c}_i(t)$ of length N . The vector element $c_{i,n}(t) \in [0, 1]$ denotes the instantaneous caching rate of file f_n , representing the proportion of the file to be cached by downloading through the fronthaul link at time t . The cache state of F-AP i with respect to (w.r.t.) f_n is denoted by $s_{i,n}(t) \in [0, S]$, which is the instantaneous number of bits stored in its caching space at time t . Assume that file f_n will be discarded by the F-AP when it is rarely requested, and the less requests for file f_n , the faster the file to be discarded from the caching space. The dynamics of the cache state can then be expressed as follows:

$$ds_{i,n}(t) = S \left(c_{i,n}(t) - e^{a-1} a^{q_{i,n}(t)} \right) dt, \quad (3)$$

where $q_{i,n}(t)$ denotes the number of requests for f_n from the serving user cluster of F-AP i at time t . Note that the incrementing rate of the cache state is represented by the caching rate $c_{i,n}(t)$, while the decrementing rate is represented by $e^{a-1} a^{q_{i,n}(t)}$, which is a descending function of $q_{i,n}(t)$ with the parameter $a \in (0, 1)$ that controls the steepness of the function. Therefore, a small request number for f_n results in a large discarding rate, which adapts to real-time requests.

In the content delivery phase, each F-AP processes the user requests during the current time slot according to the cache states of its own and the other F-APs, which can be classified into three cases. For case 1, a user that requests for file f_n associates with its default serving F-AP if the file is already cached. Otherwise, for case 2, the user seeks other adjacent F-APs that have cached f_n and associates with the one that provides the largest transmission rate [4]. For case 3, when none of the F-APs has cached the requested file, the user will then still associate with the default serving F-AP. Considering that the cache state $s_{i,n}(t)$ is a continuous variable ranging from 0 to S , here we assume that file f_n is thought to be cached by the default serving F-AP as long as $s_{i,n}(t) \geq \frac{1}{2}S$, while file f_n is thought to be cached by the other F-APs only when $s_{-i,n}(t) = S$, where $s_{-i,n}(t)$ denotes the cache state of the other possible serving F-APs.

Given the system model introduced above, the overall state of F-AP i can be represented by its channel state in (2) and cache state in (3). Let $\mathbf{h}_i = [h_{i,1}, h_{i,2}, \dots, h_{i,K}]^T$ and $\mathbf{s}_i = [s_{i,1}, s_{i,2}, \dots, s_{i,N}]^T$ denote the channel state and cache state vector of F-AP i , respectively. Then, the state vector of F-AP i can be denoted by $\mathbf{x}_i = (\mathbf{h}_i, \mathbf{s}_i)$.

The objective of this paper is to find the optimal edge caching policy to jointly minimize the request service delay and fronthaul traffic load for each F-AP.

III. THE PROPOSED DYNAMIC DISTRIBUTED EDGE CACHING SCHEME

In this section, we firstly define the cost function for each F-AP. Then, we formulate the stochastic differential game to model the caching optimization problem. Finally, a distributed mean field edge caching scheme that can adapt to time-variant user requests is proposed.

A. Cost Function

Let $J_{i,n}(t)$ denote the total cost of F-AP i for file f_n at time t , which is composed of request service delay and fronthaul traffic load.

1) *Request Service Delay*: Let $D_{i,n}(t)$ denote the delay cost of F-AP i with respect to file f_n , which is the total service delay of the requests for f_n from its serving user cluster \mathcal{U}^i during the t th time slot. Note that the request service delay refers to the total elapsed time from the file being requested to the user being successfully served, which is composed of wireless transmission delay and possible fronthaul retrieving delay.

The request service delay is expressed in (4) at the top of this page, where $\text{req}_k(t)$ denotes the requested file of

$$D_{i,n}(t) = \sum_{\substack{k \in \mathcal{U}^i \\ \text{req}_k(t)=n}} \left((C_1(t, s_{i,n}(t)) + C_3(t, s_{i,n}(t), s_{-i,n}(t))) \left(\frac{S}{R_{i,k}(t)} + \frac{S - s_{i,n}(t)}{R^F} \right) + C_2(t, s_{i,n}(t), s_{-i,n}(t)) \frac{S}{R_{-i,k}(t)} \right). \quad (4)$$

$$O_{i,n}(t) = \left(\eta_1 c_{i,n}(t) + \frac{1}{2} \eta_2 c_{i,n}^2(t) \right) + \eta q_{i,n}(t) (S - s_{i,n}(t)) (C_1(t, s_{i,n}(t)) + C_3(t, s_{i,n}(t), s_{-i,n}(t))). \quad (5)$$

user k at time t , R^F denotes the fronthaul transmission rate, and $R_{-i,k}(t)$ denotes transmission rate provided by the other serving F-AP in case 2. The occurrence conditions for the three cases introduced in Section II-B can be expressed as follows:

$$\begin{aligned} C_1(t, s_{i,n}(t)) &= H\left(s_{i,n}(t) - \frac{1}{2}S\right), \\ C_2(t, s_{i,n}(t), s_{-i,n}(t)) &= \left(1 - H\left(s_{i,n}(t) - \frac{1}{2}S\right)\right) H(s_{-i,n}(t) - S), \\ C_3(t, s_{i,n}(t), s_{-i,n}(t)) &= \left(1 - H\left(s_{i,n}(t) - \frac{1}{2}S\right)\right) (1 - H(s_{-i,n}(t) - S)), \end{aligned}$$

where $H(x) = \begin{cases} 0, & x < 0 \\ 1, & x \geq 0 \end{cases}$ is the heaviside step function.

The first term in (4) denotes the delay when the user is served by its default F-AP i , which corresponds to case 1 and 3. Hence, it is the product of the occurrence condition and the delay under the condition, which refers to the wireless transmission delay between F-AP i and user k and the fronthaul delay induced by retrieving the uncached segment of f_n from the cloud server. The second term in (4) denotes the delay when the user is served by some adjacent F-AP in case 2, and the corresponding delay refers to the wireless transmission delay brought by the other serving F-AP. Note that $s_{-i,n}(t)$ and $R_{-i,k}(t)$ can be approximated by the average states of all the other F-APs denoted by $\bar{s}_{-i,n}(t)$ and $\bar{h}_{-i,k}(t)$, which will be presented in Section III-C.

2) *Fronthaul Traffic Load*: The traffic load on the fronthaul link is induced by both content caching in the caching phase and content retrieving in the delivery phase. Let $O_{i,n}(t)$ denote the fronthaul load of F-AP i with respect to f_n at time t , and it can be expressed in (5) at the top of this page. The first term in (5) characterizes the instantaneous fronthaul load in the content caching process with η_1 and η_2 as parameters, and it is a simple quadratic function that is monotonous increasing with the caching rate $c_{i,n}(t)$. The second term in (5) is the fronthaul load generated by content retrieving when users that request for f_n are served by F-AP i , i.e., case 1 and case 3. In these two cases, F-AP i has to repeatedly retrieve the uncached file segment through the fronthaul link for each request; hence, the fronthaul load is the product of the request number, the size of the file segment to be retrieved and the occurrence condition, while η is a numerical coefficient.

Given (4) and (5), the total instantaneous cost $J_{i,n}(t)$ of

F-AP i can be represented by the weighted summation of the request service delay and the fronthaul traffic load, and it can be expressed as follows:

$$J_{i,n}(t) = \omega_1 D_{i,n}(t) + \omega_2 O_{i,n}(t), \quad (6)$$

where ω_1 and ω_2 are the coefficients to bring the two corresponding terms to the same scale. Define $\mathbf{J}_i(t) = [J_{i,1}(t), J_{i,2}(t), \dots, J_{i,N}(t)]^T$. Then, it denotes the cost vector of F-AP i at time t for all files.

B. Stochastic Differential Game Formulation

Consider the caching placement optimization in a finite time horizon $[0, T]$. As can be observed from (4) and (5), the cost function of each F-AP is determined by states of all the F-APs; hence the states and policies of F-APs have direct impacts on each other. Therefore, the caching optimization problem of F-APs can be characterized by a stochastic differential game that captures the interactive relationship among the F-APs.

The stochastic differential game is defined by a 4-tuple $(\mathcal{I}, \{\mathcal{X}_i\}_{i \in \mathcal{I}}, \{\mathcal{C}_i\}_{i \in \mathcal{I}}, \{\mathcal{J}_i\}_{i \in \mathcal{I}})$. \mathcal{I} is the set of players, which in this case are F-APs; \mathcal{X}_i denotes the state set of player i , which is the space of possible state vector \mathbf{x}_i ; \mathcal{C}_i denotes the action set of player i , which represents the set of possible caching policy vector \mathbf{c}_i ; \mathcal{J}_i denotes the cost of player i for all files over the time horizon with $\mathcal{J}_i = \mathbb{E} \left[\int_0^T \mathbf{J}_i(t) dt \right]$.

In the SDG established above, the caching optimization problem of player i can be expressed as follows:

$$\min_{\mathbf{c}_i(0 \rightarrow T)} \mathcal{J}_i(\mathbf{c}_i, \mathbf{x}_i, \mathbf{x}_{-i}), \quad (7)$$

where $\mathbf{c}_i(0 \rightarrow T)$ denotes the caching policy over the time horizon, and $\mathbf{x}_{-i} = \{\mathbf{x}_j, j \neq i, j \in \mathcal{I}\}$ denotes the states of the other $I - 1$ players.

According to the theory of dynamic programming, the overall optimum of a problem for the entire time horizon $[0, T]$ can be obtained by sequentially constructing the optimal policies of its subproblems over time period $[t, T]$ in a time-reversed manner [11]. Correspondingly, we define the value function $v_{i,n}(t)$ of player i w.r.t. f_n at time t as the minimal cost over time period $[t, T]$ as follows:

$$v_{i,n}(t) = \min_{\mathbf{c}_{i,n}(t \rightarrow T)} \mathbb{E} \left[\int_t^T J_{i,n}(r) dr \right]. \quad (8)$$

In the SDG, the set of caching policies $\{\mathbf{c}_i^*(0 \rightarrow T), i \in \mathcal{I}\}$ that minimizes the cost functions constitutes a feedback Nash

equilibrium (NE), and it is guaranteed if there exist value functions that satisfy the following backward Hamilton-Jacobi-Bellman (HJB) partial differential equations (PDEs) [12]:

$$\partial_t v_{i,n}(t) + \min_{c_{i,n}(t)} [Lv_{i,n}(t) + J_{i,n}(t)] = 0, \forall i \in \mathcal{I}, f_n \in \mathcal{F}, \quad (9)$$

where $Lv_{i,n}(t)$ denotes the differential operator of the value function.

To solve the coupled I HJB equations for each file, however, requires each F-AP to acquire the state information of all the other F-APs, which needs high computational complexity and induces enormous extra communicating overhead. Therefore, we propose to solve this challenging problem by introducing mean field game approximation.

C. The Proposed Mean Field Edge Caching Scheme

The intractability of the solution to the original SDG due to the coupled PDEs can be addressed by an approximation of the original game as a mean field game, in which individual states of all the players can be captured by a single statistical distribution [13]. Therefore, the number of the coupled PDEs can be significantly reduced and the players are able to make decisions in a distributed manner.

The establishment of a mean field game is based on four conditions [14]: (1) rationality of players; (2) continuum of players; (3) exchangeability among states of players; (4) interactions among players of the mean field type. The first condition can be guaranteed since each F-AP makes logical decisions. The second condition can be assured according to the ultra-dense deployment of F-APs. The third condition holds since the contribution of individuals become infinitesimal among the mass of players and the interchange among F-APs indices does not alter the outcome of the game. The fourth condition is assured since each F-AP interacts with the mean field instead of actual individuals in the game.

Given the properties presented above, the original SDG can be approximated as an MFG, in which each player is indistinguishable among the mass of players. Therefore, we can consider a generic player by dropping the F-AP index. The interactions between the generic player with the others in the MFG are sufficiently characterized by its own state \mathbf{x} and the statistical distribution of the mass state \mathbf{m} . $\mathbf{x} = (\mathbf{h}, \mathbf{s})$ denotes the state of the generic player. \mathbf{m} represents the mean field distribution, and its element w.r.t. file f_n can be expressed as follows:

$$m_n(t) = \lim_{I \rightarrow \infty} \frac{1}{I} \sum_{i=1}^I \delta_{\mathbf{x}_{i,n}(t)}, \quad (10)$$

where $\delta_{\mathbf{x}_{i,n}(t)}$ is the Dirac measure. Note that the mean field distribution is the probability distribution of the states of all the players in the game. The evolution of $m_n(t)$ corresponds to the Fokker-Planck-Kolmogorov (FPK) equation as follows [15]:

$$\begin{aligned} \partial_t m_n(t) = & \frac{1}{2} \sigma_h^2 \partial_{hh}^2 m_n(t) - \frac{1}{2} \alpha (\mu_h - h(t)) \partial_h m_n(t) \\ & - S \left(c_n(t) - e^{a-1} a^{q_n(t)} \right) \partial_s m_n(t). \end{aligned} \quad (11)$$

Given the mean field distribution, the average states of other players in the mean field can be approximated as the functions of $m_n(t)$ as follows:

$$\begin{cases} \bar{h}_{-,k}(t) = \int h_k(t) m_n(t) dh_k, \\ \bar{s}_{-,n}(t) = \int s_n(t) m_n(t) ds_n. \end{cases} \quad (12)$$

With the establishment of the mean field, each player can obtain its policy based solely on the local information, including its local state and the mean field distribution. Correspondingly, the objective of each player can be reformulated as a generic problem as follows:

$$\min_{c(0 \rightarrow T)} \mathcal{J}(c, \mathbf{x}, \mathbf{m}). \quad (13)$$

The solution of (13) can be obtained by solving the following HJB equation:

$$\partial_t v_n(t) + \min_{c_n(t)} [Lv_n(t) + J_n(t)] = 0, \quad (14)$$

where the differential operator $Lv_n(t)$ can be expressed as follows:

$$\begin{aligned} Lv_n(t) = & \frac{1}{2} \sigma_h^2 \partial_{hh}^2 v_n(t) + \frac{1}{2} \alpha (\mu_h - h(t)) \partial_h v_n(t) \\ & + S \left(c_n(t) - e^{a-1} a^{q_n(t)} \right) \partial_s v_n(t). \end{aligned} \quad (15)$$

Here we define the Hamiltonian function of the optimization problem as $G(t, c_n, \mathbf{x}) = Lv_n(t) + J_n(t)$. As can be observed from (14), the optimal caching policy $c_n^*(t)$ can be obtained by minimizing the Hamiltonian and concretely derived as follows:

$$c_n^*(t) = -\frac{1}{\eta_2} \left(\frac{S}{\omega_2} \partial_s v_n + \eta_1 \right). \quad (16)$$

Clearly, the solution of the HJB equation affects the FPK equation through the caching policy $c_n(t)$, while the solution of the FPK equation determines the HJB equation through the cost function $J_n(t)$. The two coupled PDEs comprehensively form the structure of the mean field: the backward HJB equation models the induction process of optimization of each individual, while the forward FPK equation models the evolution of the mean field as a whole. The solutions, i.e., $m_n(t)$ and $v_n(t)$, of the FPK-HJB equations in (11) and (14) constitute the NE of the MFG. The existence and uniqueness of the NE are guaranteed by the smoothness of the Hamiltonian [15], which can be easily verified.

Note that the primary I coupled equations have been reduced to two simple equations that can sufficiently characterize the game. Each F-AP can iteratively solve the above two equations till convergence to obtain the optimal caching policy with respect to each file. In the case that the overall cache state of F-AP i exceeds the space limit C , the files with the minimal obtained $v_n(t)$ can be chosen to meet the constraint. Correspondingly, the optimization can be executed in a distributed manner without observing the specific states of other individual F-APs.

IV. SIMULATION RESULTS

The proposed distributed edge caching scheme is verified through simulations, in which a circular area with the radius

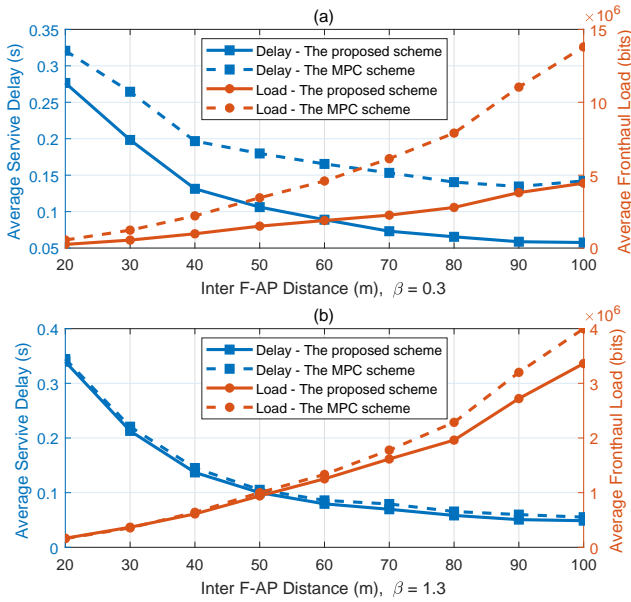


Fig. 1. Average request service delay and fronthaul traffic load versus inter F-AP distance for the proposed and MPC schemes under static user requests with $\beta = 0.3$ and $\beta = 1.3$.

of 600 m is considered. The coupled HJB-FPK equations are iteratively solved to the fixed point by utilizing the PDEPE toolbox of MATLAB. Due to the close distance between users and F-APs in ultra-dense F-RAN, a static channel model is assumed in the simulations, thus the dynamic states of the F-APs can be characterized simply by the cache states [6]. Moreover, the time horizon T is assumed to be normalized. Other parameters are set as follows: $P_i = 1$ W, $\sigma^2 = -77$ dBm, $W = 10$ MHz, $K = 100$, $S = 100$ MB, $\eta_1 = 0.0001$, $\eta_2 = 0.05$, $\eta = 0.3$, $\omega_1 = 100$, $\omega_2 = 10^{-6}$.

Assume that the probability of a user requesting file f_n follows Zipf distribution, i.e., $p_n = \frac{n^{-\beta}}{\sum_{n=1}^N n^{-\beta}}$, where the parameter β depicts the steepness of the distribution with a positive value and a larger value of β implicates a more skewed distribution of content popularity. In Fig. 1, we show the average request service delay and fronthaul traffic load versus inter F-AP distance (IFD) of the proposed scheme and the most popular caching (MPC) scheme under static user requests with $\beta = 0.3$ and $\beta = 1.3$, respectively. As shown, the average request service delay decreases with IFD, while the average fronthaul traffic load of F-APs increases with IFD. The reason for this behavior is that the F-RAN becomes sparse and the inter F-AP interference decreases as IFD increases, which result in increased wireless transmission rate and requests number of each F-AP. Moreover, Fig. 1(a) shows an obvious performance gap between the proposed scheme and the MPC scheme, while the performance gain is relatively small in Fig. 1(b). This is a result of the fact that the MPC scheme performs well only when the preference of users can be depicted by the fixed Zipf distribution. On the contrary, the proposed scheme shows satisfactory performance with different β , because it is able to capture the real-time request status even when users show less obvious preference for certain contents.

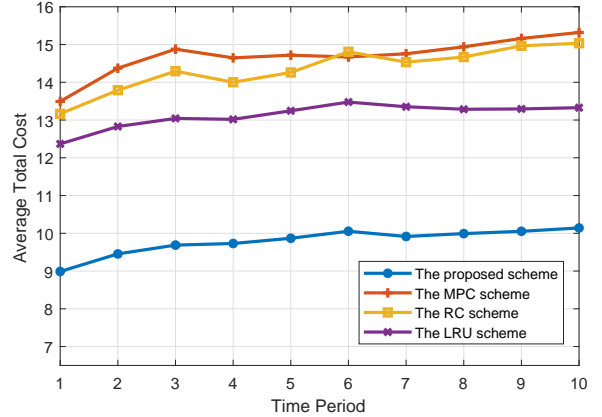


Fig. 2. Average total cost versus time period for the proposed and baseline schemes under time-variant user requests.

In Fig. 2, we show the average total cost of the proposed and baseline schemes under time-variant user requests, where the content popularity distribution is randomized in consecutive time periods. We choose the MPC, random caching (RC), and least recently used (LRU) [16] schemes as three baselines. It can be observed that the proposed scheme shows a stable performance gain over the baseline schemes. Specifically, the average total cost is reduced by 33% and 25% compared with the MPC and LRU schemes, respectively. The reason is that the MPC and RC schemes employ static caching policies without considering time-variant user requests, and the LRU scheme employs dynamic caching policies based only on local cache state. On the contrary, the proposed scheme optimizes the caching policies according to real-time requests and takes into account both local state and mean field distribution. Therefore, the proposed scheme can adapt to time-variant user requests to minimize the average total cost.

V. CONCLUSIONS

In this paper, we have proposed a dynamic distributed edge caching scheme in ultra-dense F-RAN by considering time-variant user requests. By utilizing MFG theory, the proposed scheme enables each F-AP to determine its dynamic caching policy based on local information to minimize both the request service delay and fronthaul traffic load. The simulation results have shown that the proposed scheme provides considerable performance improvement over the baseline schemes.

ACKNOWLEDGMENTS

This work was supported in part by the Research Fund of the State Key Laboratory of Integrated Services Networks (Xidian University) under grant ISN19-10, the Research Fund of the Key Laboratory of Wireless Sensor Network & Communication (Shanghai Institute of Microsystem and Information Technology, Chinese Academy of Sciences) under grant 2017002, the Hong Kong, Macao and Taiwan Science & Technology Cooperation Program of China under grant 2014DFT10290, the Ericsson and SEU Cooperation Project under grant 8504000335, the National Basic Research Program of China (973 Program) under grant 2012CB316004, and the

U.K. Engineering and Physical Sciences Research Council under Grant EP/K040685/2.

REFERENCES

- [1] Y. Ku, D. Lin, C. F. Lee, P. J. Hsieh, H. Y. Wei, C. T. Chou, and A. C. Pang, "5G radio access network design with the fog paradigm: Confluence of communications and computing," *IEEE Commun. Mag.*, vol. 55, no. 4, pp. 46–52, Apr. 2017.
- [2] N. Bhushan, J. Li, D. Malladi, R. Gilmore, D. Brenner, A. Damnjanovic, R. T. Sukhvasi, C. Patel, and S. Geirhofer, "Network densification: The dominant theme for wireless evolution into 5G," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 82–89, Feb. 2014.
- [3] M. Sheng, C. Xu, J. Liu, J. Song, X. Ma, and J. Li, "Enhancement for content delivery with proximity communications in caching enabled wireless networks: Architecture and challenges," *IEEE Commun. Mag.*, vol. 54, no. 8, pp. 70–76, Aug. 2016.
- [4] J. Li, Y. Chen, Z. Lin, W. Chen, B. Vucetic, and L. Hanzo, "Distributed caching for data dissemination in the downlink of heterogeneous networks," *IEEE Trans. Commun.*, vol. 63, no. 10, pp. 3553–3568, Oct. 2015.
- [5] A. Abboud, E. Baştuğ, K. Hamidouche, and M. Debbah, "Distributed caching in 5G networks: An alternating direction method of multipliers approach," in *Proc. IEEE 16th Int. Workshop Signal Processing Advances Wireless Commun.*, June 2015, pp. 171–175.
- [6] K. Hamidouche, W. Saad, M. Debbah, and H. V. Poor, "Mean-field games for distributed caching in ultra-dense small cell networks," in *Proc. American Control Conf.*, July 2016, pp. 4699–4704.
- [7] K. Poularakis and L. Tassiulas, "Code, cache and deliver on the move: A novel caching paradigm in hyper-dense small-cell networks," *IEEE Trans. Mobile Comput.*, vol. 16, no. 3, pp. 675–687, Mar. 2017.
- [8] H. Kim, J. Park, M. Bennis, S. L. Kim, and M. Debbah, "Ultra-dense edge caching under spatio-temporal demand and network dynamics," in *Proc. 2017 IEEE Int. Conf. Commun.*, May 2017, pp. 1–7.
- [9] A. Sadeghi, F. Sheikholeslami, and G. B. Giannakis, "Optimal and scalable caching for 5G using reinforcement learning of space-time popularities," *IEEE J. Sel. Topics Signal Processing*, vol. 12, no. 1, pp. 180–190, Feb 2018.
- [10] H. Tembine and M. Debbah, "Reacting to Interference: A mean field Stochastic Game Approach," in *Mechanisms and Games for Dynamic Spectrum Allocation*, M. L. H. . H. V. P. Tansu Alpcan, Holger Boche, Ed. Cambridge University Press, Dec. 2013, pp. 57–73.
- [11] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. Athena Scientific, 2005.
- [12] D. W. K. Yeung and L. Petrosjan, *Cooperative Stochastic Differential Games*. Springer, 2006.
- [13] C. Yang, J. Li, M. Sheng, A. Anpalagan, and J. Xiao, "Mean field game-theoretic framework for interference and energy-aware control in 5G ultra-dense networks," *IEEE Wireless Commun.*, vol. 25, no. 1, pp. 114–121, Feb. 2018.
- [14] O. Guéant, "A reference case for mean field games models," *J. De Math. Pures et Appliquées*, vol. 92, no. 3, pp. 276–294, Sept. 2009.
- [15] O. Guéant, J.-M. Lasry, and P.-L. Lions, "Mean field games and applications," in *Paris-Princeton Lectures on Mathematical Finance*. Springer, 2011, pp. 205–266.
- [16] H. Ahlehagh and S. Dey, "Video caching in radio access network: Impact on delay and capacity," in *2012 IEEE Wireless Commun. Netwk. Conf.*, Apr. 2012, pp. 2276–2281.