IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 5, 2018, accepted January 8, 2019, date of publication February 15, 2019, date of current version February 27, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2896637

Joint Cache Allocation With Incentive and User Association in Cloud Radio Access Networks Using Hierarchical Game

TRA HUONG THI LE[1], NGUYEN H. TRAN[2], (Senior Member, IEEE), PHUONG LUU VO[1,3], ZHU HAN[1,4], (Fellow, IEEE), MEHDI BENNIS[1,5], (Senior Member, IEEE), AND CHOONG SEON HONG[1], (Senior Member, IEEE)

[1]Department of Computer Science and Engineering, Kyung Hee University, Yongin 17104, South Korea
[2]School of Computer Science, The University of Sydney, Sydney, NSW 2006, Australia
[3]School of Computer Science and Engineering, International University-Vietnam National University Ho Chi Minh City, Ho Chi Minh City 7000, Vietnam
[4]Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77004-4005, USA
[5]Centre for Wireless Communications, University of Oulu, 90014 Oulu, Finland

Corresponding author: Choong Seon Hong (cshong@khu.ac.kr)

This work was supported by the National Research Foundation of Korea (NRF) funded by the South Korean Government (MSIT), under Grant NRF-2017R1A2A2A05000995.

ABSTRACT In this paper, we consider a cloud radio access network-based system consisting of one network operator (NO) and several content providers (CPs). The NO owns a cloud cache and provides caching as a service for CPs, who provide contents to users. While the NO wishes to motivate CPs to rent its cache and maximize its profit, CPs want to optimize the service performance for users and their renting utilities. Due to the time separation between cache allocation and user association problems, we model the interactions between the NO and CPs as a hierarchical game, i.e., a cache renting scheme between the NO and CPs in the cache allocation problem and the willingness of CPs in the user association problem. In the cache allocation problem, we propose a contract theory-based incentive mechanism in which the NO designs and offers an optimal contract to various types of CPs. We then formulate the user association problem as a many-to-many matching game with externalities. To solve this matching game, we propose a matching algorithm that converges to a two-sided exchange stable matching with low complexity. The simulation results demonstrate that this proposed approach is beneficial to the NO's profit and incentivize the CP to rent the cache with truthful private information. In addition, the system performance of the proposed approach in terms of the total data rate–delay tradeoff outperforms than the benchmarks.

INDEX TERMS Cloud RAN, caching as a service, cache allocation, contract theory, asymmetric information, matching game, externalities.

I. INTRODUCTION

CRAN is a promising paradigm technology for meeting the rapid increase in mobile Internet traffic. CRAN represents a combination of wireless and information technology (IT) where cloud computing is incorporated into radio access networks [1]–[3]. In CRANs, the traditional role of the base stations (BSs) is split into two parts based on their functions: a pool of baseband units (BBUs) deployed with high-performance processors and the distributed remote radio heads (RRHs) equipped with antennas, and used for compressing and forwarding signals between users and centralized BBU pool [4], [5]. As connected with RRHs through wired or wireless fronthaul links, the BBU pool redefines RAN as a software defined environment instead of the conventional hardware defined infrastructure. Therefore, more functionalities can be incorporated into the BBU pool [6].

Caching as a service is one functionality extension for CRAN, where the requested contents can be obtained from the dynamic storage deployed in the cloud cache. The use of caching at the cloud level has recently emerged as a promising technique to overcome the limitations of the backhaul link.

The associate editor coordinating the review of this manuscript and approving it for publication was Zehua Guo.

The advantages of caching at the cloud level are: 1) BBUs have greater cache sizes, larger coverage areas and more easily to scalable compared with BSs or APs; 2) BBUs are closer to the users than the content servers in the core network; and 3) RRHs do not need to be modified, instead only focusing on the basic signal transmission functionalities [6], [7].

In this paper, we consider a CRAN system with caching as a service consisting of one NO and multiple CPs. The NO is the owner of the CRAN system, which includes cloud cache, fronthaul and RRHs. The NO divides its cache space into multiple partitions and leases them to the CPs which have users but no infrastructure. One real example of such a NO is the SKT company in Korea. In 2016, SKT with Nokia first deployed a CRAN system, and has further plan to provide caching services at BBUs in the near future [8], [9]. Similar to edge caching at BSs, some CPs such as Netflix, Hulu, Sling TV, HBO Now, etc. can rent cache space from the NO to reduce the capital expenditure (CAPEX) and operating expenditure (OPEX). The NO wants to maximize its own cache leasing revenue, which depends on how the cache is allocated to the CPs as well as the corresponding payment. Differently, CPs desire to maximize their benefits while providing a better quality of service performance considering the data rate and the E2E delay of their users. The E2E delay is the sum of the backhaul delay and the wireless transmission delay. The backhaul delay is reduced by judiciously renting the cloud caches while the wireless transmission delay and data rate can be improved using an efficient user association scheme. CPs need to choose a proper cache capacity to rent as well as the corresponding payment to satisfy both the renting benefits and the service performance. Therefore, in this paper, we focus on two problems: **cache allocation and user association.**

On the one hand, if the cache allocation is known, the backhaul delay can be determined. Moreover, the backhaul delay is a part of the E2E delay, which should be minimized when making the user association decision. Thus the cache allocation's output affects the outcome of user association. On the other hand, changing the user association leads to changes in the wireless transmission delay of CPs. Because the user association and cache allocation are different in timescale and because CPs wish to minimize the E2E delay, we assume the willingness of CPs to rent the cache is proportional to the wireless transmission delay: that is, the larger the wireless transmission delay for each CP suffers, the more willingness to rent the CP is. Therefore, changing the wireless transmission delay because of the user association changes the order of willingness of CPs or the willingness difference between CPs. In addition, the cloud cache capacity is limited and the NO wants to maximize its cache leasing utility and attract more CPs to rent. Thus, changing the rent willingness of CPs changes in the cloud cache partition and the allocation to the CPs. Therefore, the cache allocation is influenced by the user association.

For the *cache allocation* problem, both the NO and CPs benefit from leasing and renting the cloud cache, respectively. The NO partitions its cache storage into virtual segments and leases each segment (partition) to the CPs. This gives the NO an opportunity to profit. In addition, the NO can reduce the backhaul usage cost when the requested file is cached at the cloud level. In turn, the CP can increase its revenue by providing a faster service for their users while reducing CAPEX and OPEX of deploying the network infrastructure for each CP. In this paper, we choose contract theory [10]–[12] as a framework to optimize the cache allocation and pricing problems. This is because in commercial caching systems, CPs can be untruthful and not reveal accurate private information in order to mislead the NO into charging them much lower prices or giving them more cache space; this is called the *asymmetric information* problem between the CPs and the NO. Under this situation, we cannot use the Stackelberg game as in previous works [13], [14] because those authors assumed that the players are truthful and reveal all private information. To deal with the asymmetric information problem, we use contract theory in this paper. *Contract theory* not only functions as the *incentive mechanism* to motivate CPs to rent the cache space and leverage their private information, but also helps the NO maximize its utility. Based on contract theory, we design the contract so that each CP is incentivized to choose the contract intended for its types which is defined as its rent willingness, thus maximizing the NO's.

"Another problem is *user association*, which determines the wireless transmission delay. In CRAN, the dense deployment of RRHs leads to severe interference. To deal with this problem, cooperative transmission (i.e., coordinated multipoint or CoMP) based on the concept of multicell transmission is a potential solution [15], [16]. In this paper, we consider a model of a cooperative transmission network where each user is cooperatively served by multiple RRHs and each RRH can serve multiple users. In addition, we realize that the user-RRH associations are affected by other peers due to intra-cluster interference. Moreover, the limited fronthaul capacity and user fairness must be considered. To obtain larger profits from users, CPs have to provide better service performance by maximizing the data rate and minimizing the E2E delay to users. Therefore, the ratio of data rate and E2E delay can be considered as the parameter for service performance of each user. In this paper, we call this ratio as the datarate - delay tradeoff. In user association problem, the objective is to maximize the social welfare of the system, defined as the sum of the tradeoffs between the data rates and the E2E delays of the users in the system. However, due to the nonconvexity of the formulated problem, the conventional exhaustive method may be impractical to solve the problem. Centralized methods can provide the optimal solution but require the high computation complexity and global control information [17]. Other methods such as noncooperative game theory have some shortcomings: distributed implementation limitation due to the requirement of knowledge of other players' actions, impractical unilateral deviations due to one-sided (or unilateral) stability notions investigation [17]. In this paper, we regard the user

association as a many-to-many matching game with *externalities*. The RRHs and users can be viewed as two sets of players to be matched with each other to maximize the utilities, considering interdependencies that exist among users due to interference. The matching game [18]–[21] can provide an adaptive and low complexity framework to solve the association problem in a self-organized manner.''

Therefore, with all the considered above problems, the main contributions of this paper are summarized as follows:

- We present a general framework to model the caching model and the E2E delay. Furthermore, we utilize a hierarchical structure to represent the interactions between the NO and CPs in both the cache allocation problem and the user association problem. Cache allocation presents a scheme to partition the cloud cache into slices and then assigns them to the CPs with the objective of maximizing the utility of the NO and incentivizing the rental use of CPs. The user association assigns the associations between the RRHs and users by considering the limited fronthaul capacity and user fairness.
- For cache allocation, we propose a contract-based model where the NO acts as a monopolist to set up the optimal cache-price contract and offers it to the CPs. The CPs are classified into different types based on their willingness to cache and their request rate. Each CP will choose the contract item that maximize its utility as compared with the alternatives. We develop an algorithm for the optimal contract design based on the sequential optimization.
- For user association, we formulate this problem as a many-to-many matching game with externalities. We propose an algorithm to obtain a two-sided exchange stable matching. We also analyze the stability, convergence, and complexity of the proposed algorithm.
- We carry out simulations to validate the effectiveness of the proposed scheme. The results show that our proposed method can guarantee good CP incentive. The results also show the effectiveness of the proposed scheme in terms of the total data rate-delay tradeoff compared with the benchmarks.

The rest of this paper is organized as follows. Some related works are presented in Section II. The system model and general problem are described in Section III followed by the hierarchical game framework in Section IV. The contract formulation is presented in Section V for the allocating cache for the CPs. In Section VI, we propose the user association algorithm based on a many-to-many matching game. The simulation results are shown in Section VII. Finally, conclusions are drawn in Section VIII.

II. RELATED WORK

A. CACHING IN CRAN

Some recent studies have been performed on caching in CRAN [22]–[25], but these works only focus on reducing the content access latency. Peng *et al.* [26] investigated the cache size allocation problem in cellular networks to maximize the user success probability (USP). Chu *et al.* [27] proposed a utility-driven cache partitioning approach for multiple content providers. A cache is partitioned into slices, with each partition being dedicated to a particular content provider. A formal proof is given in [27] that partitioning the cache yields better performance compared to sharing. The work in [28] is similar to [26] but was explored from a game-theoretic cache allocation standpoint. However, these works do not consider the commercial perspective, where the NO and CPs participate in renting and leasing cache space. Both the NO and CPs benefit from this type of being commerce and are selfish, wishing to maximize their own benefits and thus, producing a competition problem among entities.

B. CONTRACT THEORY

Contract theory is often studied to handle the information asymmetry problem and can be applied to many areas of wireless networks such as Device to Device (D2D), cognitive radio, delay tolerance networks (DTN), etc. Zhang *et al.* [29] proposed a contract theoretic approach to address the problem of incentive user participation in D2D communications. Gao *et al.* [30] proposed a framework to solve the problem of spectrum sharing in cognitive radio networks with a primary user (PU) which is a seller who offers a spectrum trading contract as (qualities, prices), and secondary users (SUs), which are buyers that select contracts to sign. In [31], contract theory was applied to solve the user incentive problem by offloading delay services for the operator in a DTN. Inspired by these studies, we model the interactions between the NO and CPs based on the contract theoretic approach to incentivize CPs to rent cache space from the NO while maximizing the NO's utility. Different from traditional contract models with two feasible contract conditions, we impose cache capacity constraints and propose an algorithm to find the optimal contract. In addition, we consider dynamic contract theory based cache allocation, which is related to the user association stage.

C. MATCHING GAME

''Matching game is a technique that provides mathematically tractable solutions for problems of matching players in two distinct sets [17], [32]. The study in [33] formulated a joint uplink (UL) and downlink (DL) user association problem that maximizes the sum-rate for UL and DL transmission of all users. They formulated the problem as a distributed two-sided iterative matching game and obtain a solution of the game. The solution of the game was guaranteed to converge and provides Pareto-optimal stable associations. In [34], the social network-aware user association in wireless small cell networks was formulated as a matching game between users and their serving nodes (base stations). The problem was decomposed into a dynamic clustering problem in which base stations were grouped into disjoint clusters based on mutual interference. Subsequently, a many-to-one user-base station matching game was carried out per cluster. The study in [35] model the interactions between the femtocell user equipments and femtocell base stations in the uplink cognitive femtocell

network. A distributed framework based on the matching game and distributed algorithms are developed to enable the cognitive femtocell network to make decisions about user association, subchannel allocation, and transmit power. Pantisano *et al.* [36] studied the problem of user cell association in a small cell network. While the preferences list of the players in [33]–[35] is fixed, the preference of players (BSs and users) in [36] are interdependent and dynamic. In addition, the authors viewed the user association as a many-to-one matching game with externalities and proposed a distributed algorithm that enables the players to self-organize into a stable matching. In contrast, our proposed technique is based on a many-to-many matching game to solve the user association problem. Because the preference lists of all players (RRHs and users) are dynamic, we present an algorithm to achieve a swap stable matching based on the concepts of swap operation and two-sided exchange stability."

III. SYSTEM MODEL

We consider one NO and a set $\mathcal{K} = \{1, 2, \ldots, K\}$ of CPs, as shown in Fig. 1. The NO owns CRAN, which includes the BBU pool, cloud cache, and a set of RRHs $\mathcal{R} = \{1, 2, \ldots, R\}$. RRH is connected to the BBU pool via the fronthaul, while the BBU pool connects to the content server through backhaul links. Cloud cache in the BBU pool helps decrease the traffic exchange between the users and the content server through limited backhaul links. The capacity of the backhaul link is limited and is represented by D^B.

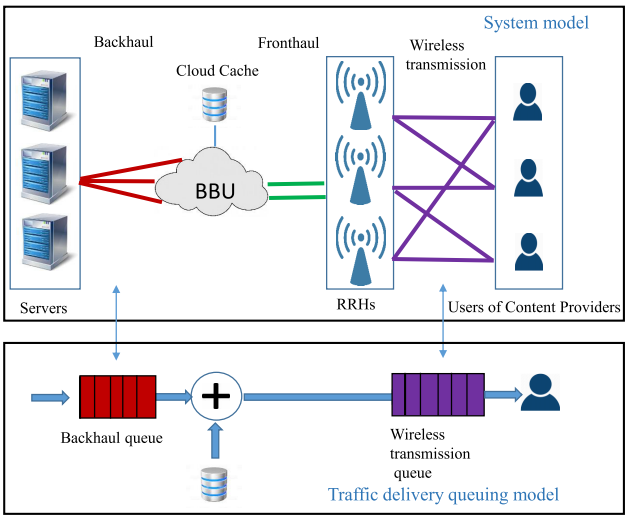


FIGURE 1. System model and corresponding traffic delivery queuing.

We assume that each CP $k \in \mathcal{K}$ owns a content server in the core network and a set $\mathcal{U}_k$ of U_k users. The total set of users in the system is denoted by $\mathcal{U} = \bigcup_{k\in\mathcal{K}} \mathcal{U}_k$. The file library of CP $k \in \mathcal{K}$ is denoted by the set $\mathcal{F}_k$ of F_k files with the same size. This is reasonable because heterogeneous contents can be divided into multiple contents chunks of the same size. We normalized the file size of 1.

TABLE 1. Table of key notation.

Description	Notation
The cache capacity of BBU	Q
Set of CPs	$\mathcal{K}$
The fraction of renting cache of CP k	β_k
The hit rate of CP k	H_k
The hit ratio of CP k	h_k
Request rate of user u	λ_u
Transmission rate of user u	c_u
Backhaul delay of user u	τ_u^B
Wireless transmission delay of user u	τ_u^W
Vector of cache fractions	$\boldsymbol{\beta}$
Vector of payments	$\boldsymbol{P}$
Utility of CP k	A_k^{CP}
Utility of the NO	A^{NO}
Set of users	$\mathcal{U}$
Set of RRHs	$\mathcal{R}$
Quota of user u	q^u
Quota of RRH r	q^r
Utility of user u	S_u^{RA}
Utility of RRH r	$S_r^{RA,R}$
Current matching	Υ
Overall social welfare	$S^{RA}(\Upsilon)$

A. CACHING MODEL

The key notations used in this paper are listed in Table 1. The request for file $f \in \mathcal{F}_k$ is modeled as an inhomogeneous Poisson point process with rate η_{fk}, which is the sum of requests for file f of all users of CP k. Based on the Che's approximation [37], [38], we have $h_{fk} = 1 - e^{\eta_{fk}T_k} \approx \eta_{fk}T_k$, where h_{fk} is the cache hit probability of file $f \in \mathcal{F}_k$ and T_k is the characteristic time of cache partition of CP k, which can be considered as timer in TTL caching. In addition, we suppose the cache partition in which CP k rents a fraction β_k of the cloud cache space. Thus, we have $\sum_{f\in\mathcal{F}_k} h_{fk} = \beta_k Q$, and therefore,

$$T_k = \frac{\beta_k Q}{\sum_{f\in\mathcal{F}_k} \eta_{fk}}. \tag{1}$$

Thus, the hit rate of CP k can be expressed as:

$$H_k = \sum_{f\in\mathcal{F}_k} \eta_{fk} h_{fk} \approx \sum_{f\in\mathcal{F}_k} \eta_{fk}^2 T_k = \frac{\sum_{f\in\mathcal{F}_k} \eta_{fk}^2 \beta_k Q}{\sum_{f\in\mathcal{F}_k} \eta_{fk}}. \tag{2}$$

The cache hit probability h_{fk} of file $f \in \mathcal{F}_k$ is the probability that a request for file f results in a hit [27]. Therefore, we consider the hit rate of CP k as the average number of requests resulting in a hit in the set of files $\mathcal{F}_k$ of CP k. Correspondingly, the hit ratio of CP k, which is defined as the ratio between the hit requests and total requests, can be given by $h_k = \frac{H_k}{\sum_{f\in\mathcal{F}_k} \eta_{fk}}$. We can see that the hit rate H_k and hit ratio h_k depend on the fraction β_k of cache space rented by CP k. In addition, to guarantee a hit ratio $h_k \leq 1$, we should have $\beta_k \leq \frac{(\sum_{f\in\mathcal{F}_k} \eta_{fk})^2}{\sum_{f\in\mathcal{F}_k} \eta_{fk}^2 Q}$.

B. TRANSMISSION MODEL

Requested content can be sent from the content server or cloud cache. We model the traffic delivery process as a queueing

system, as shown in Fig. 1. Through out the paper, we assume that the requests within each queue are served in a FIFO manner and the buffer size is infinite.

1) BACKHAUL DOWNLINK TRAFFIC

When a requested file is not cached in cloud cache, the file has to be fetched from the content server through the backhaul. We assume that traffic arrives at the backhaul toward user $u \in \mathcal{U}_k$ according to a Poisson process with a request rate of $\lambda_u(1-h_k)$. Let $\alpha = \sum_{k\in\mathcal{K}}\sum_{u\in\mathcal{U}_k}(\lambda_u(1-h_k))$ be the total the traffic through the backhaul links. Then, we assume that the traffic is evenly distributed to m active backhauls to the BBUs with probability $\frac{1}{m}$. Therefore, the mean incoming traffic rate routed to each active backhaul is $\frac{\alpha}{m}$. We also assume that the expected data rates of the backhauls are constant during the user association process. We model the traffic delivery in the backhaul as an M/M/1 queuing system. Therefore, the backhaul delay of user $u \in \mathcal{U}_k$ is as follows:

$$\tau_u^B = \frac{\lambda_u(1-h_k)}{D^B - \frac{\alpha}{m}}. \tag{3}$$

2) WIRELESS TRANSMISSION TRAFFIC

Denote by γ_u the received signal to interference plus noise ratio (SINR) at user u, which is given by [22]

$$\gamma_u = \frac{\sum_{r\in\mathcal{R}} p_{ur} g_{ur} x_{ur}}{I_{inter} + I_{intra} + \sigma^2}, \tag{4}$$

where p_{ur} and g_{ur} are the transmit power and channel coefficient from RRH r to user u, respectively. x_{ur} is the association indicator, which is equal to 1 if user u associates with RRH r, and equal to 0, otherwise. The first component $I_{inter} = \sum_{r\in\mathcal{R}} p_{ur} h_{ur}(1-x_{ur})$ in the denominator is the inter interference, and the second component $I_{intra} = \sum_{r\in\mathcal{R}}\sum_{u'\in\mathcal{U}} p_{u'r} h_{u'r} x_{ur} x_{u'r}$ is the intra interference. σ^2 characterizes the noise spectral density at each user's receiver. The achievable rate of user $u \in \mathcal{U}$ is expressed as

$$c_u = W \log(1+\gamma_u) \tag{5}$$

where W is the bandwidth. Here, the channel gain reflects the slow fading including pathloss and shadowing. In addition, according to Burke's Theorem [39], the arrival process of wireless transmission is a Poisson function with rate λ_u. Therefore, the wireless transmission queue for user $u \in \mathcal{U}$ can be regarded as an M/M/1 queue, and the wireless transmission delay for user $u \in \mathcal{U}$ is expressed as

$$\tau_u^W = \frac{\lambda_u}{c_u - \lambda_u}. \tag{6}$$

Therefore, the end-to-end delay of delivering traffic for user $u \in \mathcal{U}_k$ is given by $\tau_u^B + \tau_u^W$. We can see that the end-to-end delay of user u is impacted by the cache capacity that CP k rents and the RRHs associated with user u. In addition, to guarantee stability of the queueing system, we should have $c_u > \lambda_u$. Moreover, to account for the limited fronthaul, the maximum number of users that can connect with RRH r is limitted, i.e.,

$$\sum_{k\in\mathcal{K}}\sum_{u\in\mathcal{U}_k} x_{ur} \le q^r. \tag{7}$$

IV. HIERACHICAL GAME FRAMEWORK

Here, we consider the interaction between the NO and CPs through two problems: the cache allocation and user association problems. Even though two processes, cache allocation and user association affect the E2E delay, they differ in time scale: the user association is determined at a much faster timescale than that of cache allocation. Both the NO and CPs are selfish in leasing and renting cache. The NO wants to attract more CPs to rent its cache spaces and the NO further maximizes its profits by offering the optimal cache partition and corresponding payment, whereas the CPs also wish to maximize their utility by obtaining a large amount of cache space with a low price. In addition, CPs want to provide the best service to their users in terms of the data rate and the E2E delay. They aim to maximize the tradeoff between the data rate and the E2E delay, which is determined by the association between RRHs and users. Specifically, after *selecting the cache partition*, each CP can estimate the *backhaul delay* for each its user. We note that the backhaul delay is a part of the *end-to-end delay*, which is an element of the *objective of RRH-user association*. Therefore, there is an indirect effect of the cache allocation results on the user association. On the other hand, depending on the user association, each CP can estimate the *wireless transmission delay* of each of its users. When the wireless transmission delay of one CP is high, the *willingness* to rent cache of this CP is high, because the E2E delay is even higher if this CP does not rent cache. In other words, the greater the delay the CP suffers due to wireless transmission, the greater its willingness to rent cache space. Since the limited capacity cache space is shared among CPs, the NO needs to determine a cache partition by fully considering the willingness of CPs in order to attract more CPs to rent. When the willingness of the CPs changes, the NO is required to modify the allocation of cache space to the CPs and corresponding payment so that the NO can maximize its utility and provide incentive to the CPs: the CP with higher renting willingness should have more cache space. There also exist indirect effects of user association results on the cache allocation results.

We use an example to explain the influence of user association on the cache allocation. Suppose there are 3 CPs A, B, C who rent the cache space from the NO. In the first time slot, the cache size percentage allocated to the three CPs are 50%, 20%, 30%, respectively. This allocation corresponds to the backhaul delay of A, B, C of 10, 11, 12, respectively. Based on the given backhaul delay, the user association is determined to maximize the social welfare. Suppose, the sum of the wireless transmission delay after user association of the three CPs are 9, 8, 6, respectively. Based on the wireless transmission delay, we have that the order of willingness

of the three CPs is A, B, C, where A has the highest willingness to rent the cache and C has the lowest. Therefore, the cache allocation in the second time slot should be in the same order as the willingness, such as: 45%, 35%, 20% for A, B, C, respectively. The scenario under investigation, therefore, allows two levels of competition and one cyclic dependency, as described above.

The competitions and cyclic dependency can be illustrated utilizing the hierarchical framework shown in Fig. 2, where two different game formulations are adopted to investigate these inter-linked problems. Especially, cache allocation is formulated using a contract model in which the NO calculates the fraction of cache space leased to each type of CP and the corresponding payment. Meanwhile, the user association is formulated as a many-to-many matching game with externalities. At the beginning of each time slot, the NO determines an optimal contract given the user association results in the last time slot. Then the NO will broadcast the optimal contract to the CPs. After evaluation of the contract, the CP sends feedback to the NO regarding the contract type. After getting the feedback from the CPs, the NO will assign the cache space to the CPs. In one time slot, the cache allocation is fixed. Based on the cache allocation, the users and RRHs will determine their coordination in order to maximize the social welfare of the system.

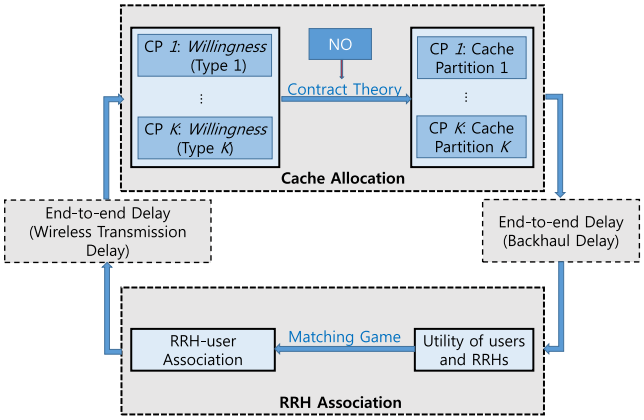


FIGURE 2. Illustration of the hierarchical game framework.

In the following, the contract cache allocation is studied from a theoretical viewpoint in Section V, and the matching game based user association will be presented in Section VI.

V. CONTRACT THEORY BASED CACHE ALLOCATION

It is known that the cache capacity of the NO is limited. In addition, if the CPs rent cache space at the cloud level, the NO can decrease the cost of downloading a file from the server through a backhaul link. Thus, the NO needs to identify an incentive mechanism to attract CPs to rent and maximize its utility. Furthermore, CPs also want to maximize their utilities. However, the NO does not know what type of CPs are present, related to the willingness to cache and the request rate. This makes it difficult for the NO to decide the optimal cache allocation and corresponding payment, which results in information asymmetry between the NO and CPs. To deal with this problem, we implement the contract theory because this method can incentivize CPs to reveal the accurate information. Specifically, the NO will design a menu of contracts to define the percentage of its total cache space leased to the CP and the corresponding payment. In this section, we present the utilities of the NO and CPs, the formulation of the contract problem, and the design of the optimal contract.

A. UTILITY MODEL FOR CACHE ALLOCATION

1) UTILITY OF CP K

The utility of CP k is defined as

$$A_k^{CP}(\beta_k, P_k) = \omega_k H_k - P_k = \omega_k \frac{\sum_{f\in\mathcal{F}_k} \eta_{fk}^2}{\sum_{f\in\mathcal{F}_k} \eta_{fk}} \beta_k Q - P_k, \quad (8)$$

where $\omega_k > 0$ and is proportional to the wireless transmission delay $\sum_{u\in\mathcal{U}_k}(\tau_u^B + \tau_u^W)$ which is given by the user association. The utility of CP k is the difference between the benefit of CP k from renting cache and the price P_k charged by the NO. We further define $\theta_k = \omega_k \frac{\sum_{f\in\mathcal{F}_k} \eta_{fk}^2}{\sum_{f\in\mathcal{F}_k} \eta_{fk}}$ as the renting willingness of CP k. We also let θ_k denote the type of CP k. We assume there are total K different types, and we denote the set of types $\boldsymbol{\Theta} = \{\theta_1, \theta_2, \ldots, \theta_K\}$. Without loss of generality, we assume $\theta_1 < \ldots < \theta_k < \ldots < \theta_K$. The NO does not have exact information for the individual θ_k of every CP k.

2) UTILITY OF THE NO FROM PARTITIONING AND LEASING CACHE TO THE CPS

The NO is the CRAN owner who owns the BBUs, fronthaul and RRHs. To transmit the requested file to the users of the CPs, the NO also needs to rent the backhaul links to retrieve files from the core network to the BBUs. However, the backhaul usage cost of the NO depends on the cache allocation among CPs at the BBUs. This is because when a requested file is cached at the BBUs, the file can be transmitted directly to the users. The utility of the NO is determined by the difference between the payment from leasing the cache to all CPs and the backhaul usage cost. Thus, the utility of the NO from partitioning and leasing cache to the CPs is expressed as

$$A^{NO}(\boldsymbol{\beta}, \boldsymbol{P}) = \sum_{k\in\mathcal{K}} (P_k - \varphi_k(\beta_k)), \quad (9)$$

where the backhaul usage cost, $\varphi_k(\beta_k)$, is a convex and decreasing function of the cache partition β_k; and $\boldsymbol{\beta} = \{\beta_k, \forall k \in \mathcal{K}\}, \boldsymbol{P} = \{P_k, \forall k \in \mathcal{K}\}$ are the cache allocation and payment vectors, respectively.

The selfish and rational NO not only wants to maximize its utility but also applies different strategies to different types of CPs with different willingness to rent cache. In the cache trading market, the private information of CPs is unknown by the NO. Some CPs can claim wrong information willingness and/or request rates so that the allocated cache partition

capacity is higher or the price charged is lower. Thus, it is difficult for the NO to know exactly what type of a CP is.

It is well known that contract theory is an useful technique in modeling the incentive mechanism under asymmetric information. The contract design in this paper can be categorized as *adverse selection* case, i.e., the type of the CP is unknown to the NO. The NO acts as a monopolist in the market and designs the contract entries for various CPs to maximize its total utility, where the contract $\{\beta_k(\theta_k), P_k(\theta_k)\}$ is assigned for CP type θ_k. If CP k accepts the contract, then this CP will receive cache partition β_k and pay an amount P_k to the NO. In the following, the interactions of the NO and CPs are formulated as a game based on the framework of contract theory.

B. CONTRACT FORMULATION

To incentivize CP type k to rent cache space, it must be individual rational (IR), i.e.,

$$\text{IR: } A_k^{CP}(\beta_k, P_k) = \theta_k \beta_k Q - P_k \geq 0, \quad \forall k \in \mathcal{K}. \tag{10}$$

The contracts should bring a non-negative utility to the CP, which motivates the CP to actively participate in the trade. In addition, type k CP would prefer to choose the contact item (β_k, P_k) rather than contract item design for other types. Thus the CP should be incentive compatible (IC), i.e.,

$$\begin{aligned}\text{IC: } & A_k^{CP}(\beta_k, P_k) \geq A_k^{CP}(\beta_j, P_j) \\ & \theta_k \beta_k Q - P_k \geq \theta_k \beta_j Q - P_j, \quad \forall k, j \in \mathcal{K},\ k \neq j. \end{aligned} \tag{11}$$

The IC condition ensures that the CP k will accept the contract (β_k, P_k) designed for it rather than choosing other contracts $(\beta_j, P_j), k \neq j$. These two constraints guarantee that the optimal contract can provide participation incentive for the users. In addition to the IC and IR constraints, the NO will need to design the contract such that the expected cache size leased meets the cache size limit of the BS, i.e.,

$$\sum_{k \in \mathcal{K}} \beta_k \leq 1. \tag{12}$$

The optimal contract design can be formulated as the NO's total utility maximization problem, i.e.,

$$\begin{aligned}(CA): \ & \max_{\boldsymbol{\beta}, \boldsymbol{P}} A^{NO}(\boldsymbol{\beta}, \boldsymbol{P}), \\ & s.t.: \ (10),\ (11),\ (12), \\ & \min\left(1, \frac{(\sum_{f \in \mathcal{F}_k} \eta_{fk})^2}{\sum_{f \in \mathcal{F}_k} \eta_{fk}^2 Q}\right) \geq \beta_k \geq 0, \quad \forall k \in \mathcal{K}. \end{aligned} \tag{13}$$

C. CONDITIONS FOR FEASIBLE CONTRACT

In this section, we analyze the feasibility of the contract design, which is formulated in (13).

Lemma 1: For any feasible contract $(\boldsymbol{\beta}, \boldsymbol{P})$, $\beta_i \geq \beta_j$ *if and only if* $\theta_i \geq \theta_j$.

Proof: We refer to [10] for the proof. This Lemma simply proves that the NO must provide more cache capacity to the CPs with higher types, which is the monotonicity property of contract design. □

Lemma 2: For any feasible contract $(\boldsymbol{\beta}, \boldsymbol{P})$, *the utilities of CPs must satisfy*

$$0 < A_1^{CP} < \ldots < A_k^{CP} < \ldots < A_K^{CP}. \tag{14}$$

Proof: We refer to [10] for the proof. □

Thus, the higher type CPs gain more utility than the lower type CPs. From the IC constraint and the two lemmas above, we can easily deduce the following. If a CP selects a contract designed meant for a high type, even though it will receive more cache space, the profit of the hit rate cannot compensate for the payment to the NO. Moreover, if a CP selects a contract intended for a low type, this CP receives a smaller cache space although it pays less to the NO. The CP can receive the maximum utility if and only if it selects the contract that best fits with its type. Thus, we can guarantee that the contract is truthfully self-revealed.

Theorem 1 (Sufficient and Necessary Conditions for Contract Feasibility): A contract $(\boldsymbol{\beta}, \boldsymbol{P})$ *is feasible if and only if all of the following three conditions hold:*

(a): $0 \leq \beta_1 \leq \beta_2 \leq \ldots \leq \beta_K$;

(b): $\theta_1 \beta_1\, Q - P_1 \geq 0$;

(c): $\theta_k(\beta_k - \beta_{k-1}) \leq P_k - P_{k-1} \leq \theta_{k-1}(\beta_k - \beta_{k-1}), \forall k \in \{2, 3, \ldots, K\}$.

The proof can be found in Appendix A.

D. OPTIMALITY OF CONTRACT

The optimal contract can be obtained either by solving problem (13) directly or by applying a sequential optimization aproach. In this part, we adopt a sequential optimization approach. First, we derive the best payment $\boldsymbol{P}^*$ given a fixed cache allocation $\boldsymbol{\beta}$, and then we derive the best cache allocation $\boldsymbol{\beta}^*$ for the optimal contract.

Lemma 3: Given the cache allocation $\boldsymbol{\beta}$, *then the unique optimal price sastifies:*

$$\begin{aligned} P_1^* &= \theta_1 \beta_1 Q, \\ P_k^* &= P_{k-1}^* + \theta_k(\beta_k Q - \beta_{k-1} Q). \end{aligned} \tag{15}$$

The proof can be found in Appendix B.

From (15), we can conclude that

$$P_k^* = \theta_1 \beta_1 Q + \sum\nolimits_{i=1}^{k} Z_i, \quad \forall k \in \{1, 2, \ldots, K\}, \tag{16}$$

where

$$Z_i = \begin{cases} 0, & \text{if } i = 1, \\ (\beta_i Q - \beta_{i-1} Q)\theta_i, & \text{if } i = 2, 3, \ldots, K. \end{cases}$$

Based on Lemma 3, we can simplify problem (13) as

$$\begin{aligned} \max_{\{\boldsymbol{\beta}\}} A^{NO} &= \sum_{k \in \mathcal{K}} (\theta_1 \beta_1 Q + \sum_{i=1}^{k} Z_i - \varphi_k) \\ s.t.\ & (12), \\ & \beta_1 \leq \beta_2 \leq \ldots \leq \beta_K, \\ & \min\left(1, \frac{(\sum_{f \in \mathcal{F}_k} \eta_{fk})^2}{\sum_{f \in \mathcal{F}_k} \eta_{fk}^2 Q}\right) \geq \beta_k. \end{aligned} \tag{17}$$

In order to solve this problem, one standard approach is to leave out the monotonicity condition and then determine whether the obtained solution satisfies this condition.

After removing the monotonicity condition, we can represent the objective function of problem (17) as:

$$\max_{\{\boldsymbol{\beta}\}} A^{NO} = \sum_{k\in\mathcal{K}} A_k^{NO}$$
$$s.t.: (12),$$
$$\min\left(1, \frac{(\sum_{f\in\mathcal{F}_k}\eta_{fk})^2}{\sum_{f\in\mathcal{F}_k}\eta_{fk}^2 Q}\right) \geq \beta_k, \quad \forall k \in \mathcal{K}, \tag{18}$$

where $A_k^{NO} = \theta_k\beta_k Q + (K-k)\Delta_k - \varphi_k$ and $\Delta_k = (\theta_k - \theta_{k+1})\beta_k Q, \forall k < K$, and $\Delta_k = 0$ when $k = K$. Obviously, all $A_k^{NO}, \forall k \in K$ are convex functions of β_k. The Langrangian of (18) is:

$$L = \sum_{k\in\mathcal{K}} A_k^{NO} - \eta\left(\sum_{k\in\mathcal{K}}\beta_k - 1\right) - \sum_{k\in\mathcal{K}}\nu_k(\beta_k - o_k), \tag{19}$$

where η and ν_k are the Langrange multipliers, $o_k = \min\left(1, \frac{(\sum_{f\in\mathcal{F}_k}\eta_{fk})^2}{\sum_{f\in\mathcal{F}_k}\eta_{fk}^2 Q}\right)$. Applying the KKT condition, $\overline{\beta}_k^*$, $k \in \mathcal{K}$ are solutions of

$$\begin{cases} (A_k^{NO})' - \eta - \nu_k = 0, & \forall k \in \mathcal{K}, \\ \nu_k(\beta_k - o_k) = 0, & \forall k \in \mathcal{K}, \\ \eta(\sum_{k\in\mathcal{K}}\beta_k - 1) = 0, & \end{cases}$$

where $(A_k^{NO})'$ is the first-order derivative of A_k^{NO} with respect to β_k.

Furthermore, we need to check whether these solutions satisfy the monotonicity condition. If $\overline{\beta}_k^*$ satisfies the monotonicity condition, it can be regarded as our desired optimal quality β_k^*. Otherwise, we need to make some adjustments based on the following proposition.

Proposition 1: Let $A_1^{NO}(\beta)$ and $A_2^{NO}(\beta)$ be concave functions on β. If $\overline{\beta}_1^* \geq \overline{\beta}_2^*$ where $\overline{\beta}_1^* = \underset{\beta_1}{argmax}\, A_1^{NO}(\beta_1)$ and $\overline{\beta}_2^* = \underset{\overline{\beta}_2}{argmax}\, A_2^{NO}(\beta_2)$, where

$$\{\beta_1^*, \beta_2^*\} = \underset{\beta_1,\beta_2}{argmax} \sum_{i=1}^{2} A_i^{NO}(\beta_i) \quad s.t.\ \beta_1 \leq \beta_2.$$

Proof: We refer to [30] for the detailed proof of Proposition 1. □

Proposition 1 can be extended to a more general form: if $\overline{\beta}_1^* \geq \overline{\beta}_2^* \geq \ldots \geq \overline{\beta}_K^*$ where $\overline{\beta}_i^* = \underset{\beta_k}{argmax}\, A_k^{NO}(\beta_k)$, then $\beta_1^* = \beta_2^* = \ldots = \beta_K^*$ where $\{\beta_k^*\} = \underset{x_i}{argmax} \sum_{k=1}^{K} A_k^{NO}(\beta_k)\ s.t\ \beta_1 \leq \beta_2 \leq \ldots \leq \beta_K$.

We denote a subsequence of $\{\overline{\beta}_k^*\}$, say $\{\overline{\beta}_i^*, \overline{\beta}_{i+1}^*, \ldots, \overline{\beta}_j^*\}$, as an infeasible subsequence, if $\overline{\beta}_i^* \geq \overline{\beta}_{i+1}^* \geq \ldots \geq \overline{\beta}_j^*$. For example, in a cache allocation $\{\overline{\beta}_k^*\} = \{0.04, 0.16, 0.16, 0.12, 0.32, 0.2\}$, there are two feasible subsequences, i.e., $\{\overline{\beta}_2^*, \overline{\beta}_3^*, \overline{\beta}_4^*\}$ and $\{\overline{\beta}_5^*, \overline{\beta}_6^*\}$. According to Proposition 1, the adjusted values satisfy $\beta_i^* = \beta_{i+1}^* = \ldots = \beta_j^*$. Moreover, $\beta_i^*, \beta_{i+1}^*, \ldots, \beta_j^*$ maintains the capacity constraints

$$\beta_i^* + \beta_{i+1}^* + \ldots + \beta_j^* = \overline{\beta}_i^* + \overline{\beta}_{i+1}^* + \ldots + \overline{\beta}_j^*.$$

Therefore, $\beta_i^* = \beta_{i+1}^* = \ldots = \beta_j^* = \frac{\overline{\beta}_i^* + \overline{\beta}_{i+1}^* + \overline{\beta}_j^*}{j-i+1}$

Substituting the feasible allocation $\{\beta_k^*\}$ into (16), we obtain the corresponding optimal price P_k^*:

$$P_k^* = \theta_1\beta_1^* Q + \sum_{i=1}^{k} Z_i^* \tag{20}$$

for any $k \in \{1, 2, \ldots, K\}$, where

$$Z_i^* = \begin{cases} 0, & if\ i = 1, \\ (\beta_i^* Q - \beta_{i-1}^* Q)\theta_i, & if\ i = 2, 3, \ldots, K. \end{cases}$$

Note that, using the sequential optimization approach, the original problem in (13) has been significantly simplified by decreasing the number of constraints and variables. In addition, based on Lemma 3, there is no gap between the solution by using the sequential optimization approach and a solution attained by solving (13) directly. The contract algorithm is illustrated in Algorithm 1.

Algorithm 1 Optimal Contract Algorithm

INPUT: $Q, K, \boldsymbol{\Theta} = \{\theta_1, \theta_2, \ldots, \theta_K\}$
OUTPUT: $(\boldsymbol{\beta}, \boldsymbol{P})$
Step 1. Cache partition **for** $k = 1, \ldots, K$ **do**
  set $A_k^{NO} = \theta_k\beta_k Q + (K-k)S\Delta_k - \varphi_k$
  $\overline{\beta}_k^*, k \in \mathcal{K}$ is the solution of $\begin{cases} (A_k^{NO})' - \eta - \nu_k = 0, \\ \nu_k(\beta_k - o_k) = 0 \\ \eta(\sum_{k\in\mathcal{K}}\beta_k - 1) = 0, \end{cases}$
end
while $\overline{\boldsymbol{\beta}}^*$ *is not feasible* **do**
  find an infeasible subsequence $\{\overline{\beta}_i^*, \overline{\beta}_{i+1}^*, \ldots, \overline{\beta}_j^*\}$
  set $\beta_i^* = \beta_{i+1}^* = \ldots = \beta_j^* = \frac{\overline{\beta}_i^* + \overline{\beta}_{i+1}^* + \overline{\beta}_j^*}{j-i+1}$
end
Step 2. Payment
for $k = 1, \ldots, K$ **do**
  set $Z_i^* = \begin{cases} 0, & \text{if } i = 1, \\ (\beta_i^* Q - \beta_{i-1}^* Q)\theta_i, & \text{if } i = 2, 3, \ldots, K. \end{cases}$
  set $P_k^* = \theta_1\beta_1^* Q + \sum_{i=1}^{k} Z_i^*$
end

VI. USER ASSOCIATION AS A MANY-TO-MANY MATCHING GAME WITH EXTERNALITIES

In this section, given a cache partition among the CPs, we focus on solving the association of RRHs with

users to maximize the total social welfare by solving the problem:

$$(RA): \max_{\boldsymbol{x}} S^{RA} = \sum_{u\in\mathcal{U}} \frac{c_u}{\tau_u^B + \tau_u^W}$$
$$s.t: (7),$$
$$\sum_{r\in\mathcal{R}} x_{ur} \le q^u, \quad \forall u \in \mathcal{U},$$
$$x_{ur} = \{0,1\}, \quad \forall u \in \mathcal{U}, \ \forall r \in \mathcal{R},$$
$$\boldsymbol{\beta} \quad \text{is given}, \tag{21}$$

where $\boldsymbol{x} = \{x_{ur}, \forall u \in \mathcal{U}, \forall r \in \mathcal{R}\}$ is the matrix of association indicator. The numerator in the objective function is the achievable rate of user u and the denominator is the total delay that user u experiences, which is the sum of the backhaul delay τ_u^B and the transmission delay τ_u^W. Given cache partition $\boldsymbol{\beta}$ from Section V, each user u can calculate the backhaul delay τ_u^B. The backhaul delay of one user depends on the cache space capacity of the CP to which this user belongs. As a result, the cache allocation affects to the user association. To this end, a utility matching algorithm is proposed. In this section, we first introduce the definition of the matching game and utility, and then describe in detail the algorithm used to obtain a stable matching.

A. DEFINITION AND UTILITY FUNCTION

We consider the set of users $\mathcal{U}$ and the set of RRHs $\mathcal{R}$ as two disjoint sets of selfish and rational players aiming to maximize their own benefits. Specifically, if RRH r is assigned to user u, then we say user u and RRH r are matched with each other and form a matching pair. A matching is defined as an assignment of RRH in $\mathcal{R}$ to users in $\mathcal{U}$, formally presented as follows

Definition 1: Given two disjoint sets $\mathcal{U}$ of users and $\mathcal{R}$ of RRHs, a many-to-many matching Υ is a mapping from the set $\mathcal{U} \cup \mathcal{R} \cup \{0\}$ into the set of all subsets of $\mathcal{U} \cup \mathcal{R} \cup \{0\}$ such that, for every $u \in \mathcal{U}$ and $r \in \mathcal{R}$:

1) $\Upsilon(u) \subseteq \mathcal{R}$
2) $\Upsilon(r) \subseteq \mathcal{U}$
3) $\mid \Upsilon(u) \mid \le q^u$
4) $\mid \Upsilon(r) \mid \le q^r$
5) $u \in \Upsilon(r) \Leftrightarrow r \in \Upsilon(u)$.

Condition 1) states that each user is matched with a subset of RRHs, and condition 2) implies that each RRH is matched with a subset of users. We assume that the number of RRHs being mapped for each user is no larger than q^u, and similarly the number of users being mapped for each RRH is no larger than q^r, taking into account the limited fronthaul link, which reflect on the condition 3) and 4), respectively. The condition 5) shows that if u matches with r, r matches with u.

The utility of user u can be defined as the ratio between the data rate and the E2E delay, which shows the tradeoff of data rate delay of user u. The utility of user u can be present as

$$S_u^{RA} = \frac{c_u}{\tau_u^B + \tau_u^W}, \quad \forall u \in \mathcal{U}. \tag{22}$$

The utility of RRH r is defined as the sum of the utilities of the users connecting to this RRH and can be presented as

$$S_r^{RA,R} = \sum_{u\in\mathcal{U}} S_u^{RA} x_{ur}, \quad \forall r \in \mathcal{R}. \tag{23}$$

The total social welfare of the system is the sum of data rate delay tradeoff of all users in the system and can be present as

$$S^{RA} = \sum_{u\in\mathcal{U}} S_u^{RA}. \tag{24}$$

B. PREFERENCE LISTS

Both users and RRHs desire to obtain a high utility. According to the utility function, each player can compute a preference list over the players of the other set. The preference list of one user is an ordering of potential RRHs, in which the RRHs that give this user a higher utility, will be higher ranked. Similarly, the preference list for an RRH is an ordering of potential users, in which the users that provide this RRH a higher utility, will be higher ranked. Specifically, for any user $u \in \mathcal{U}$, its preference u over the set of RRHs can be described as follows. For any two RRHs $r, r' \in \mathcal{R}, r \ne r'$ and any two matchings $\Upsilon, \Upsilon', r \in \Upsilon(u), r' \in \Upsilon'(u)$:

$$(r, \Upsilon) \succ_u (r', \Upsilon') \Leftrightarrow S_u^{RA}(\Upsilon) > S_u^{RA}(\Upsilon'). \tag{25}$$

where $S_u^{RA}(\Upsilon)$ is the utility of user u when user u associates with r in matching Υ and $S_u^{RA}(\Upsilon')$ is the utility of user u when user u associates with r' in matching Υ'. Similarly, for any RRH $r \in \mathcal{R}$, its preference over the set of users can be explained as follows. For any two users $u, u' \subseteq \mathcal{U}, u \ne u'$, and any two matchings $\Upsilon, \Upsilon', u \in \Upsilon(r), i' \in \Upsilon'(r')$:

$$(u, \Upsilon) \succ_r (u', \Upsilon') \Leftrightarrow S_r^{RA,R}(\Upsilon) > S_r^{RA,R}(\Upsilon'), \tag{26}$$

where $S_r^{RA,R}(\Upsilon)$ is the utility of RRH r when RRH r associates with use u in matching Υ and $S_r^{RA,R}(\Upsilon')$ is the utility of RRH r when RRH r associates with use u' in matching Υ'. (26) implies that r prefers user u to u' only when r can obtain a higher utility from u.

It is easy to see that the utility of each user is a function of the interference from the other users associating with the same RRHs. Therefore, the preference lists of the users not only depend on the RRHs to which the user is matched with, but also the other users associated with the same RRHs. Similarly, the preference lists of the RRHs change under different matching states. In other words, there is interdenpendency of the preference lists of users and RRHs. Therefore, the matching game formulated above is a many-to-many matching with externalities. Influenced by peer effects [19], the outcome of this matching game heavily depends on the dynamic interactions between the users. Due to the peer effect between players, we introduce the concept of a two-sided exchange stable matching based on the operations of the matching results, namely swap matching as given below.

Definition 2: Let Υ be a matching with $r \in \Upsilon(u)$, $r' \in \Upsilon(u')$ and $r' \notin \Upsilon(u), r \notin \Upsilon(u')$. Let Υ' denote a modified matching where two pairs swap their match,

i.e., $r' \in \Upsilon'(u)$, $r \in \Upsilon'(u')$ *and* $r \notin \Upsilon'(u)$, $r' \notin \Upsilon'(u')$. *Swap matching occurs when*

$$\begin{aligned}
&1)\ \forall s \in \{u, u'\}, \quad S_s^{RA}(\Upsilon') \geq_s S_s^{RA}(\Upsilon),\\
&2)\ \exists s \in \{u, u'\}, \quad S_s^{RA}(\Upsilon') >_s S_s^{RA}(\Upsilon),\\
&3)\ \forall s \in \{\Upsilon(r'), \Upsilon(r)\}|\{u, u'\}, \quad S_s^{RA}(\Upsilon') \geq_s S_s^{RA}(\Upsilon). \quad (27)
\end{aligned}$$

Swap matching occurs when the utility of any player involved in current matching will not decrease, which is shown in condition (1) and at least one player's utility will increase, which is shown in condition (2). Condition (2) avoids looping between equivalent matchings where the utilities of all involved players are unchanged. In addition, we have modified the definition in [19] via the condition (3). This condition means that the utilities of the users matching with the RRHs involved in the swap also increase. It is noted that one of the user pairs involved in the swap can be a ''hole'', representing an open spot in an RRH, thus allowing for single-user movement to available vacancies. Similarly, one of the RRHs r involved in the swap can be a ''hole''. However, in the conditions in (2), the utilities of all ''holes'' and players in the opposite set matched with the ''holes'' are not considered. Through multiple swap operations, we show how dynamic preferences of different players are associated with each other, and the matching games's externalities are well handled. The players keep executing approved swap operations so as to reach a stable status, also known as a two-sided exchange stable matching, defined as below.

Definition 3: A matching Υ *is two-sided exchange stable (2ES) if no swap matching can occur.*

C. STABLE MATCHING

With this definition of stability, we introduce two RRH-user matching algorithms RRHA-1 to obtain a 2*ES* matching. This algorithm is an extended version of the many-to-one matching algorithms proposed in [19] and was inspired by the work in [40] and [41]. Different from the many-to-one matchings, we consider the constraints $|\ \Upsilon(u)\ | \leq q^u$ and $|\ \Upsilon(r)\ | \leq q^r$ in the RRHA-1. The key idea of RRHA-1 is to keep considering approved swap matchings among the players so as to reach a 2ES matching. Algorithm RRHA-1 has two phases. In phase 1, the initial states are set up. RRHs and users are randomly matched with each other to satisfy the constraints $|\ \Upsilon(u)\ | \leq q^u$ and $|\ \Upsilon(r)\ | \leq q^r$. In phase 2, each user keeps looking for the other user and the available vacancies of RRHs for which they can swap their matching RRHs (swap matching conditions are satisfied) and then update the current matching. This iteration stops when no more swaps occur.

Algorithm 2 RRHA-1

INPUT: τ_u^B, q^u, q^r
OUTPUT: $\boldsymbol{x} = \{x_{u,r}, \forall u \in \mathcal{U}, \forall r \in \mathcal{R}\}$
Phase 1: Random matching
- Each user and RRH is randomly matched with another subject according to $|\ \Upsilon(u)\ | \leq q^u$ and $|\ \Upsilon(r)\ | \leq q_r$.
Phase 2: Swap matching
while *there exist swap matching* **do**
  for *every matching user* $u \in \mathcal{U}$ **do**
    -search for the $\mathcal{U}\backslash\{u\}$ or an open spot $\mathcal{O}$ of RRH's available vacancies for the swap matching pair (u, u') along with $r \in \Upsilon(u)$, $r' \in \Upsilon(u')$, or swap matching pair $(u, \mathcal{O})$,
    - **if** *swap occurs* **then**
      - update $\Upsilon = \Upsilon'$
      else
        u keep its matches
      end
    end
  end
end

D. STABILITY, CONVERGENCE, COMPLEXITY

In the following, we analyze the performance of the RRHA-1 algorithm based on user association. More specifically, we prove the convergence, define and analyze the local maxima, and prove the stability.

Definition 4: Social welfare is the matching Υ *for which there exists no matching* Υ'*, obtained from* Υ *by swapping two users such that* $S^{RA}(\Upsilon') > S^{RA}(\Upsilon)$ *where* $S^{RA}(\Upsilon')$ *and* $S^{RA}(\Upsilon)$ *are the total social welfare of the system when the matching of the system are* Υ *and* Υ'*, respectively.*

Theorem 2: The proposed RRHA-1 algorithm converges to a 2ES matching after a limited number of swap operations. The proof can be found in Appendix C.

Lemma 4: All local maxima of $S^{RA}(\Upsilon)$ *are two-sided exchange stable.*

The proof can be found in Appendix D.

However, not all 2*ES* matchings obtained by RRHA-1 are local maxima of $S^{RA}(\Upsilon)$. For example, a swap might not occur because the utility of one user will decreases although the utility of another user increases a lot; then the overall utility increases. If the swap is forced, then a one-sided exchange stable matching occurs at the expense of the utility of one user.

With regard to the computational complexity of the algorithm RRHA-1, we present the analysis as follows. In algorithm RRHA-1, for swap matching, a number of iterations are performed to reach the final matching. In each iteration, the users search for swapping users or an open spot. Thus the complexity of the swap matching phase depends on the numbers of iterations and attempts at swap matching in each iteration. When $Uq^u = Rq^r$, each player remains fully matched before and after every swap matching; thus, for any swap matching, there are two users and two RRHs. For each user u, there are $q^u(R - q^u)$ possible RRH couples to be swapped. Moreover, each RRH has a quota of q^r. Thus for each user, there are $q^u(R - q^u)q^r$ possible swaps. Since there are U users, there are $\frac{1}{2}Uq^u(R - q^u)q^r$ swap matchings in each iteration. With a given number of total iterations of I', we have a computational complexity of

$\frac{1}{2}I'Uq^u(R-q^u)q^r \approx \frac{1}{2}I'URq^uq^r$, because in practice, the total number of RRHs R is usually significantly bigger than the quota q^u. Therefore, we have that the computation complexity of algorithm RRHA-1 is $O(I'URq^uq^r)$.

VII. SIMULATION RESULTS

In this section, we present the simulation results to evaluate our proposed framework. In this paper, the cache capacity has the unit of files. We consider a cloud cache with capacity in the range from 1×10^5 to 15×10^5 files in which each file has normalized size of 1. Each CP serves a random number of requests between 4×10^6 and 5.6×10^6. We adopt homogeneous settings for users in the same CP, and the users in the same CPs have the same file access pattern. File popularities for the users of the CPs follow the Zipf distribution in which the exponent characterizing the distribution is varied from 0.5 to 0.6. The request rate of a user, which is the average number of requests of this user per unit of time, is randomly chosen. The request rate for file f of user u of CP k being the product of request rate of this user and the popularities pattern of file f. In addition, we assume the RRHs are deployed at fixed locations. We consider a log-distance path loss model given by Oo *et al.* [42]. The power density of the noise is -174 dBm/Hz. Bandwidth $= 60$ KHz, transmitting power $= 1$ watt.

A. PERFORMANCE OF CONTRACT BASED CACHE ALLOCATION

To evaluate the proposed contract algorithm, which is called as **C-Scheme**, three other cache allocation schemes are implemented as follows.

- **N-Scheme**: Cache allocation is considered without information asymmetry. In this case, the NO only needs to consider the IR conditions and cache limitation constraints when it designs the contract.
- **DC-Scheme**: This scheme follows the decompose-and-compare algorithm in [43].
- **O-Scheme**: In this scheme, the payment of CP type k is defined as $P_k = \gamma_k(\beta_k)^2$. Thus, the utility of CP type k is $A_k^{CP}(\beta_k) = \theta_k\beta_k Q - \gamma_k(\beta_k)^2$. We set the cache allocation in this scheme as the same as that in **C-Scheme**. To verify the unique optimal price with given cache allocation, the payment is deploy deployed under condition $(A_k^{CP})' = 0$, which is showed in Lemma 3.

The unit for the value of the utility is the unit of money.

Fig. 3 depicts the utility of the NO and six CPs in four different cache allocation schemes. From this figure, we can observe that the N-scheme gives the NO the highest utility compared with three other schemes. This is because the contract designed by the NO in the N-scheme is required to satisfy only the IR conditions, which means that the NO in the N-scheme has more opportunities to make profit than in the C-Scheme. Compared with the DC-scheme and O-scheme, the utility of the NO in the C-scheme is higher because the cache allocation and payment in the C-scheme are the optimal solutions of (13). In contrast, the utilities of the CPs in the DC-scheme and O-scheme are higher than in the C-scheme and all six CPs in the N-scheme have a utility of zeros.

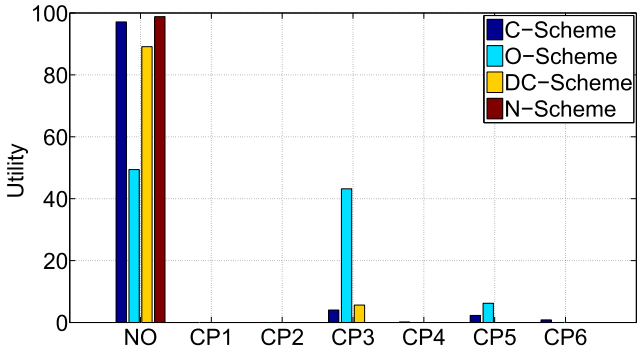


FIGURE 3. Numerical results for comparisons of the utilities of the NO and CPs achieved by four different schemes.

Fig. 4 represents the utility of each CP received by selecting the different contracts for different types. Here the types of CP1, CP2, CP3, CP4, CP5, CP6 are 1, 2, 4, 6, 5, 3, respectively in which type 6 is the highest and type 1 is the lowest. We can see the higher-type CPs receive higher utilities and the lowest-type CP receives zero utility. These results reflect the IR conditions in the contract design. In addition, each type of CP can achieve maximum utility when selecting the right contract design for this type. For example, CP 3, which is classified as type 6, receives a utility of 4.0336 when it signs the type 6 contract. This utility is higher than the utility it can get when it signs other types of contracts, such as type-1, type-2 or type-4 contracts. Another example is CP 1, which is type-1. CP 1 receives zero utility when it signs the type-1 contract and negative utility when it signs other contracts. These examples verify the IC conditions in contract design.

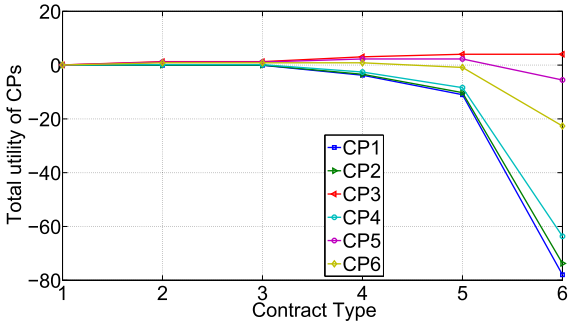


FIGURE 4. Numerical results for the total utility of CPs when they sign different contracts.

Fig. 5 shows the impact of the CRAN cache capacity on the utility of the NO. There are six CPs with different numbers of files in the libraries and different popularity patterns. It can be observed that the utilities of the NO in four schemes except the DC-scheme increase when the cache capacity grow. It is because the InP gains more profit from leasing the cache with a larger cache capacity. However, in the DC-scheme, first, the NO's utility increases when the cache capacity changes from 10^5 to 9×10^5 files and then decreases when the cache

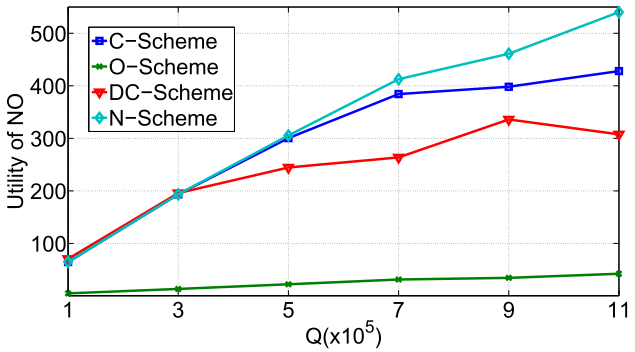


FIGURE 5. Numerical results for the impact of the cache capacity Q on the utility of the NO.

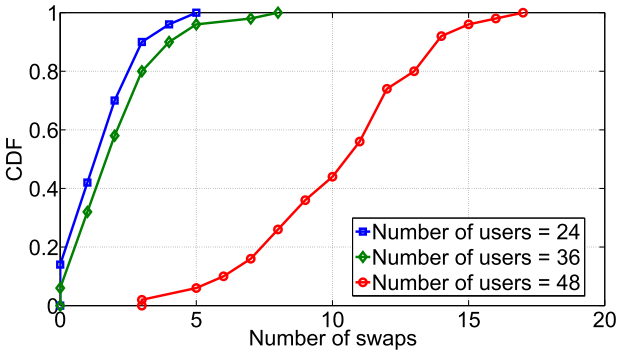


FIGURE 7. Numerical results for Distribution of the total number of swap operations in RRHA-1.

capacity changes from 9×10^5 to 11×10^5 files. This can be explained as follows. When the cache capacity Q increases, the range of β_k is narrower due to the relation between the hit ratio, cache capacity and cache fraction. Furthermore, in DC-scheme, the NO designs the contract by selecting the highest utility contract among the set of candidate ones. The narrower range of β_k leads to the reduction of the number of candidate contracts. Therefore, the increase of cache capacity causes the loss of the NO's utility in the DC-scheme. In addition, in various cache capacity settings, the proposed C-Scheme and N-Scheme allow the NO to obtain the highest utilities, followed by the O-Scheme and the DC-Scheme.

Fig. 6 presents the variation of the NO's utility when the number of CPs is varied from 6 to 16. In this figure, we assume the cache capacity of the NO is $Q = 15 \times 10^5$ and all CPs have the same the number of files in the libraries, the popularity patterns and the number of requests. Fig. 5 witnesses the increasing of the NO's utility when the number of CPs increases. This results from competition among the CPs to rent the cache space at the cloud.

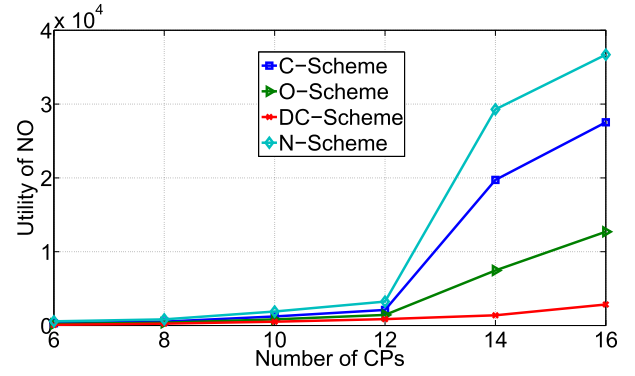


FIGURE 6. Numerical results for the impact of the number of CPs on the utility of the NO.

B. PERFORMANCE OF MATCHING GAME-BASED USER ASSOCIATION AND THE PROPOSED HIERARCHICAL GAME FRAMEWORK

Fig. 7 shows the CDF of the number of swap operations required for the RRHA-1 algorithm to converge when the number of RRHs is 12 and the quota of each user and RRH are 3 and 12, respectively. The speed of convergence decreases when the number of users increases. Fig. 3 further reflects the low computational complexity of the proposed RRHA-1. For example, when the number of users is 48, the RRHA-1 converges after 17 swap operations.

Fig. 8 illustrates the CDF of total data rate-delay tradeoff v.s. the maximal number of users that can associate with the same RRH q^r. It is showed that when q^r changes from 4 to 5 or from 5 to 6, the total tradeoff increases because the users can select better RRHs to maximize their utilities. However, when q^r change from 5 to 6, there is no improvement in the total tradeoff because when more users can be associated with one RRH, more intra-interference occurs between the user suffers. We conclude that the system is limited by interference.

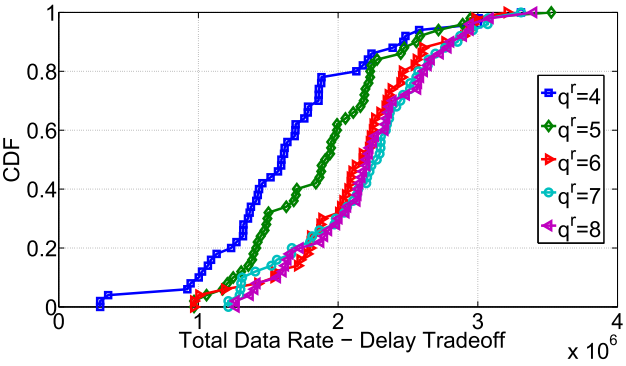


FIGURE 8. Numerical results for Total social welfare v.s. the quota of each RRH.

Furthermore, to verify the effectiveness of our proposed scheme, we compare the proposed scheme (CA+RRHA) with other schemes: i) contract theory based cache allocation and random RRH-user association (CA+Random), ii) no cache at the cloud cache and random RRH-user association (Random), and iii) no cache at the cloud cache and RRHA1 algorithm based RRH-user association (RRHA-1). The cache capacity is $Q = 3 \times 10^5$, the number of RRH is 12, the number of CPs are 6, the quota of each user and RRH are 2 and 6. Fig. 9 shows the total datarate-delay tradeoff when the request

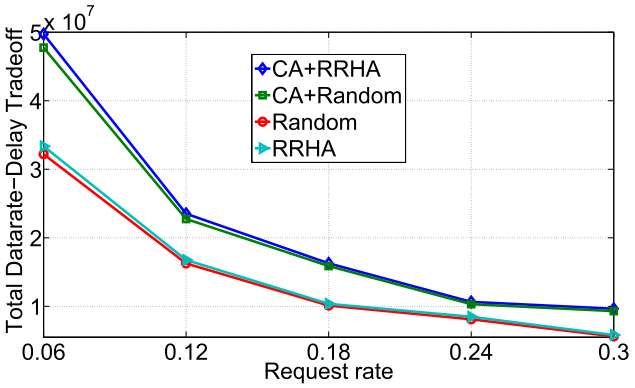


FIGURE 9. Numerical results for Total tradeoff when changing the request rate.

rate is varied. The unit of request rate is the average number of requests by all users of CPs during one unit of time. In this simulation, we assume that every user of CPs has the same number of requests per unit of time. However, the number of requests for different files of users of different CPs are different because the file popularities of CPs are different from each other. From Fig. 9, we can observe that when there are more requests to serve, the total datarate-delay tradeoff decreases. This is due to the increase in delay. In addition, when the request rate increases, the differences between the schemes using and not using the cloud cache are smaller. This is due to the fact that the growth of the request rate leads to a decline in the hit ratio, which results in increase in delay. This result shows the reduction of the effects of caching on the tradeoff of the data rate and delay with the request rate. In Fig. 10, we assume that the total numbers of requests of each CP are fixed. Therefore, when the number of users grows, the request rate of one user of each CP decreases, resulting in a decreasing in delay of each user. This is a reason why in Fig. 10 we can observe an increase in total datarate-delay tradeoff in all four schemes with the number of users. Both Fig. 9 and Fig. 10 show that the proposed scheme outperforms the other three benchmarks in terms of the total tradeoff, which indicates the effectiveness of our proposed scheme. When the number of users is 48 and the request rate is 0.06, the total tradeoffs of the proposed scheme are two times and 1.5 times higher than that of non-caching schemes, respectively. As compared with random user association and contract based allocation, our proposed scheme can provide competitive total tradeoff.

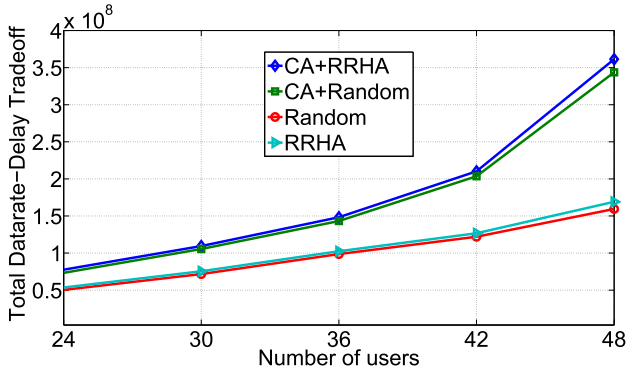


FIGURE 10. Numerical results for Total tradeoff when changing the number of users per CP and number of CPs.

VIII. CONCLUSION

In this paper, we investigate the cache allocation and user association problems for the CRAN system with caching as a service. For the cache allocation problem, we model the commercial caching system as a monopoly market, where the NO owns the cloud cache and RRHs and offers a cache partition related contract to various types of CPs while maximizing its own profits. We first formulate the optimal contract problem under an information asymmetric scenario. Afterward, the algorithm for designing the optimal contract is presented. Based on the cache allocation, we consider the user association problem to maximize the social welfare, defined as the sum of the tradeoff of the data rates and the E2E delay of users. We formulate the problem of user association as a many-to-many matching game with externalities and propose a matching algorithm to achieve a two-sided exchange stable matching within a limited number of iterations. The properties of the proposed algorithm also are discussed. Numerical results verify the effectiveness of the proposed scheme in incentivizing CPs to rent the cache and reveal their private information. The utility of the NO is guaranteed to be maximal. In addition, the simulation results demonstrate that our proposed scheme is able to improve at least two times of social welfare as compared with the non-caching schemes, even when the number of user is small. Our results also show that swap matching based proposed scheme is competitive with the random based user association. We hope the gap between swap matching and random based user association larger by the future work.

APPENDIX A
PROOF OF THE THEOREM 1

Before proving Theorem 1, we consider the three following Lemmas.

Lemma 5: Given that the IC constraint holds, for the optimal contract under incomplete information, the IR constraint can be reduced by

$$\theta_1\beta_1 Q - P_1 \geq 0. \tag{28}$$

Proof: With definition of types : $\theta_1 < \ldots < \theta_k < \ldots < \theta_K$, we have $\theta_k\beta_k Q - P_k \geq \theta_k\beta_1 Q - P_1 \geq \theta_1\beta_1 Q - P_1 \geq 0$. □

Lemma 6: If Local Downward Incentive Contraints (LDIC) are satisfied for all user type θ_k, $k \in \mathcal{K}$, i.e.,

$$\theta_k\beta_k Q - P_k \geq \theta_k\beta_{k-1} Q - P_{k-1} \tag{29}$$

then IC contraints will hold for any $h \leq k$, $h \in \mathcal{K}$, i.e.,

$$\theta_k\beta_k Q - P_k \geq \theta_k\beta_h Q - P_h. \tag{30}$$

Proof: We have two LDIC as follows:

$$\theta_k\beta_k Q - P_k \geq \theta_k\beta_{k-1}Q - P_{k-1}. \quad (31)$$

$$\theta_{k-1}\beta_{k-1}Q - P_{k-1} \geq \theta_{k-1}\beta_{k-2}Q - P_{k-2}. \quad (32)$$

We have $\theta_k > \theta_{k-1}$, so the inequality in (32) becomes

$$\begin{aligned}\theta_k(\beta_{k-1} - \beta_{k-2})Q &\geq \theta_{k-1}(\beta_{k-1} - \beta_{k-2})Q \\ &\geq P_{k-1} - P_{k-2}. \quad (33)\end{aligned}$$

Additionally, (31) is equivalent to

$$\theta_k(\beta_k - \beta_{k-1})Q \geq P_k - P_{k-1}. \quad (34)$$

Summing (33) and (34), we have:

$$\theta_k\beta_k Q - P_k \geq \theta_k\beta_{k-2}Q - P_{k-2}. \quad (35)$$

Therefore, if the LDIC holds for type-k-1 CP, the incentive constraint with respect to type k-2 holds. This process can be extended downward from type k-2 to 1 CPs, which prove that all the downward incentive constraints hold. In view of the random selection of θ_k, we have completed the proof. □

Lemma 7: If Local Upward Incentive Contraints (LUIC) are satisfied for all user type θ_k, $k \in \mathcal{K}$, i.e.,

$$\theta_k\beta_k Q - P_k \geq \theta_k\beta_{k+1}Q - P_{k+1}, \quad (36)$$

then IC contraints will be satisfied for any $l \geq k$, $l \in \mathcal{K}$,

$$\theta_k\beta_k Q - P_k \geq \theta_k\beta_l Q - P_l. \quad (37)$$

Proof: Proof is similar to Lemma 6. □

Proof for sufficient conditions of Theorem 1: (a) is implied from Lemma 1. (b) is implies from Lemma 5. The left inequality in (c) is derived from the LUIC $\forall k \in \mathcal{K}$. The right inequality in (c) is derived from the LDIC for all $k \in \mathcal{K}$.

Proof for necessary conditions of Theorem 1: (a) is implied from Lemma 1. (b) is the same as the necessary IR constraint for the lowest CP type. (c) is derived from the necessary IC constraint $\forall k \in \mathcal{K}$.

APPENDIX B
PROOF OF THE LEMMA 3

Proof: Let us proceed contradiction. Given the fixed cache allocation, the utility of the NO is decided by $\sum_{k=1}^{K} q_k P_k$. Suppose that there exists another feasible payment $\{P'_k, \forall k\}$ that has better solution than $\{P^*_k, \forall k\}$ in (15). Thus, there is at least one price $P'_k > P^*_k$ for one type θ_k. If $k = 1$, then $P'_1 > P^*_1$. Since $P^*_1 = \theta_1\beta_1 Q$, then $P'_1 > \theta_1\beta_1 Q$, which violates the IR constraints for type θ_1. If $k > 1$, since $\{P'_k, \forall k\}$ must satisfy the LDIC: $\theta_k\beta_k Q - P'_k \geq \theta_k\beta_{k-1}Q - P'_{k-1}$ or $P'_k \leq P'_{k-1} + \theta_k(\beta_k Q - \beta_{k-1}Q)$. By substituting $P^*_k = P^*_{k-1} + \theta_k(\beta_k Q - \beta_{k-1}Q)$ into this equality, we have $P'_{k-1} \geq P^*_{k-1}$. By the induction method, we have $P'_1 > P^*_1$, which violates the IR constraint for type θ_1. So we have (15). □

APPENDIX C
PROOF OF THE THEOREM 2

We prove the Theorem 2 based on three following Lemmas.

Lemma 8: Any swap leads to improvement in total social welfare, i.e., $S^{RA}(\Upsilon') > S^{RA}(\Upsilon)$.

Proof: We consider the arbitrary user u_{i_k} the possible cases. In the first case, this user is not associated with RRHs involved in the swap matching, and so the utility of this user does not change. In the second case, this user is one of two users involved in the swap matching, and the utility of this user remains the same or increases under conditions (1) and (2) of Definition 2. In the third case, this user is associated with one RRH that participating in the swap matching, and so the utility of its is not decreasing due to the condition (3). □

Lemma 9: The proposed RRHA-1 algorithm is guaranteed to converge to the final matching after a limited number of swap operations.

Proof: Since the number of users and RRHs are finite, we can find that the number of possible swaps for users are finite. From Lemma 8, the total system social welfare increases after swapping and has an upper bound. Therefore, there exists a swap operation after which no more swapping occurs and the social welfare stops increasing. RRHA-1 then converges to final matching Υ^*. □

Lemma 10: If the RRHA-1 converges to a matching Υ^*, then Υ^* is a 2ES matching

Proof: When the while loop in phase 2 of the algorithm RRHA-1 is terminated, no user can find another with which do swap. Thus, the matches of one user must be the best choice for it in the current matching. So no user has the desire to change from the current matching. Hence, the final result matching is 2 ES matching. □

APPENDIX D
PROOF OF THE LEMMA 4

Proof: Suppose the total utility of matching Υ is the local maxima of $S_{RA}(\Upsilon)$. The Υ is not a 2ES matching, which means that there exists another swap that can increase the overall utility of current matching, this contradicts the assumption that Υ is a local maximum. Thus, Υ is stable. □

REFERENCES

[1] S. Bi, R. Zhang, Z. Ding, and S. Cui, "Wireless communications in the era of big data," *IEEE Commun. Mag.*, vol. 53, no. 10, pp. 190–199, Oct. 2015.
[2] M. Peng, Y. Sun, X. Li, Z. Mao, and C. Wang, "Recent advances in cloud radio access networks: System architectures, key techniques, and open issues," *IEEE Commun. Surveys Tuts.*, vol. 18, no. 3, pp. 2282–2308, 3rd Quart., 2016.
[3] A. Checko *et al.*, "Cloud RAN for mobile networks—A technology overview," *IEEE Commun. Surveys Tuts.*, vol. 17, no. 1, pp. 405–426, 1st Quart., 2015.
[4] Y. Cai, F. R. Yu, and S. Bu, "Dynamic operations of cloud radio access networks (C-RAN) for mobile cloud computing systems," *IEEE Trans. Veh. Technol.*, vol. 65, no. 3, pp. 1536–1548, Mar. 2016.
[5] D. Liu, S. Han, C. Yang, and Q. Zhang, "Semi-dynamic user-specific clustering for downlink cloud radio access network," *IEEE Trans. Veh. Technol.*, vol. 65, no. 4, pp. 2063–2077, Apr. 2016.

[6] J. Tang and T. Q. S. Quek, "The role of cloud computing in content-centric mobile networking," *IEEE Commun. Mag.*, vol. 54, no. 8, pp. 52–59, Aug. 2016.

[7] L. Pu, L. Jiao, X. Chen, L. Wang, Q. Xie, and J. Xu, "Online resource allocation, content placement and request routing for cost-efficient edge caching in cloud radio access networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 8, pp. 1751–1767, Aug. 2018.

[8] SKT. *Nokia Implement 'World First' Commercial Cloud RAN*. Accessed: Sep. 12, 2016. [Online]. Available: https://www.telegeography.com/products/commsupdate/articles/2016/09/12/skt-nokia-implement-world-first-commercial-cloud-ran/

[9] *SK Telecom's Network Evolution Strategies (3)—Ran Architecture Evolution Strategy*. Accessed: Oct. 27, 2014. [Online]. Available: https://www.netmanias.com/en/post/blog/6682/c-ran-fronthaul-lte-sdn-nfv-sk-telecom-samsung/sk-telecom-s-network-evolution-strategies-3-ran-architecture-evolution-strategy

[10] P. Bolton and M. Dewatripont, *Contract Theory*. Cambridge, MA, USA: MIT Press, 2005.

[11] J.-J. Laffont and D. Martimort, *The Theory of Incentives: The Principal-Agent Model*. Princeton, NJ, USA: Princeton Univ. Press, 2009.

[12] Y. Zhang and Z. Han, *Contract Theory for Wireless Networks*. Cham, Switzerland: Springer, 2017.

[13] J. Li, H. Chen, Y. Chen, Z. Lin, B. Vucetic, and L. Hanzo, "Pricing and resource allocation via game theory for a small-cell video caching system," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 8, pp. 2115–2129, Aug. 2016.

[14] F. Shen, K. Hamidouche, E. Baştuğ, and M. Debbah. (2016). "A Stackelberg game for incentive proactive caching mechanisms in wireless networks." [Online]. Available: https://arxiv.org/abs/1609.02596

[15] B. Niu, Y. Zhou, H. Shah-Mansouri, and V. W. Wong, "A dynamic resource sharing mechanism for cloud radio access networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 12, pp. 8325–8338, Dec. 2016.

[16] T. X. Tran and D. Pompili, "Dynamic radio cooperation for user-centric cloud-RAN with computing resource sharing," *IEEE Trans. Wireless Commun.*, vol. 16, no. 4, pp. 2379–2393, Apr. 2017.

[17] Y. Gu, W. Saad, M. Bennis, M. Debbah, and Z. Han, "Matching theory for future wireless networks: Fundamentals and applications," *IEEE Commun. Mag.*, vol. 53, no. 5, pp. 52–59, May 2015.

[18] A. E. Roth and M. Sotomayor, "Two-sided matching," *Handbook Game Theory Econ. Appl.*, vol. 1, pp. 485–541, 1992.

[19] E. Bodine-Baron, C. Lee, A. Chong, B. Hassibi, and A. Wierman, "Peer effects and stability in matching markets," in *Proc. Int. Symp. Algorithmic Game Theory*. Cham, Switzerland: Springer, 2011, pp. 117–129.

[20] Z. Han, Y. Gu, and W. Saad, *Matching Theory for Wireless Networks*. Springer, Apr. 2017.

[21] Z. Han, D. Niyato, W. Saad, T. Basar, and A. Hjørungnes, *Game Theory in Wireless and Communication Networks: Theory, Models, and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2012.

[22] Z. Zhao, M. Peng, Z. Ding, W. Wang, and H. V. Poor, "Cluster content caching: An energy-efficient approach to improve quality of service in cloud radio access networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1207–1221, May 2016.

[23] T. X. Tran, A. Hajisami, and D. Pompili, "Cooperative hierarchical caching in 5G cloud radio access networks," *IEEE Netw.*, vol. 31, no. 4, pp. 35–41, Jul./Aug. 2017.

[24] D. Chen, S. Schedler, and V. Kuehn, "Backhaul traffic balancing and dynamic content-centric clustering for the downlink of fog radio access network," in *Proc. IEEE 17th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Edinburgh, U.K., Jul. 2016, pp. 1–5.

[25] M. Chen, W. Saad, C. Yin, and M. Debbah, "Echo state networks for proactive caching in cloud-based radio access networks with mobile users," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3520–3535, Jun. 2017.

[26] X. Peng, J. Zhang, S. H. Song, and K. B. Letaief, "Cache size allocation in backhaul limited wireless networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kuala Lumpur, Malaysia, May 2016, pp. 1–6.

[27] W. Chu, M. Dehghan, D. Towsley, and Z.-L. Zhang, "On allocating cache resources to content providers," in *Proc. 3rd ACM Conf. Inf.-Centric Netw.*, Kyoto, Japan, Sep. 2016, pp. 154–159.

[28] S. Hoteit, M. El Chamie, D. Saucez, and S. Secci, "On fair network cache allocation to content providers," *Comput. Netw.*, vol. 103, pp. 129–142, Jul. 2016.

[29] Y. Zhang, L. Song, W. Saad, Z. Dawy, and Z. Han, "Contract-based incentive mechanisms for device-to-device communications in cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 10, pp. 2144–2155, Oct. 2015.

[30] L. Gao, X. Wang, Y. Xu, and Q. Zhang, "Spectrum trading in cognitive radio networks: A contract-theoretic modeling approach," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 4, pp. 843–855, Apr. 2011.

[31] Y. Li, J. Zhang, X. Gan, L. Fu, H. Yu, and X. Wang, "A Contract-based incentive mechanism for delayed traffic offloading in cellular networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 8, pp. 5314–5327, Aug. 2016.

[32] S. Samarakoon, M. Bennis, W. Saad, and M. Latva-Aho, "Dynamic clustering and on/off strategies for wireless small cell networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 3, pp. 2164–2178, Mar. 2016.

[33] S. Sekander, H. Tabassum, and E. Hossain, "Decoupled uplink-downlink user association in multi-tier full-duplex cellular networks: A two-sided matching game," *IEEE Trans. Mobile Comput.*, vol. 16, no. 10, pp. 2778–2791, Oct. 2017.

[34] M. I. Ashraf, M. Bennis, W. Saad, M. Katz, and C.-S. Hong, "Dynamic clustering and user association in wireless small-cell networks with social considerations," *IEEE Trans. Veh. Technol.*, vol. 66, no. 7, pp. 6553–6568, Jul. 2017.

[35] T. LeAnh *et al.*, "Matching theory for distributed user association and resource allocation in cognitive femtocell networks," *IEEE Trans. Veh. Technol*, vol. 66, no. 9, pp. 8413–8428, Sep. 2017.

[36] F. Pantisano, M. Bennis, W. Saad, S. Valentin, and M. Debbah, "Matching with externalities for context-aware user-cell association in small cell networks," in *Proc. Globecom Workshops (GC Wkshps)*, Atlanta, GA, USA, Dec. 2013, pp. 4483–4488.

[37] M. Dehghan, L. Massoulie, D. Towsley, D. Menasche, and Y. Tay, "A utility optimization approach to network cache design," in *Proc. 35th Annu. IEEE Int. Conf. Comput. Commun. (INFOCOM)*, Apr. 2016, pp. 1–9.

[38] N. C. Fofack, M. Dehghan, D. Towsley, M. Badov, and D. L. Goeckel, "On the performance of general cache networks," in *Proc. 8th Int. Conf. Perform. Eval. Methodol. Tools*, Bratislava, Slovakia, Dec. 2014, pp. 106–113.

[39] D. P. Bertsekas, R. G. Gallager, and P. Humblet, *Data Networks*, vol. 2. New Jersey, NJ, USA: Prentice-Hall, 1992.

[40] T. Sanguanpuak, S. Guruacharya, N. Rajatheva, M. Bennis, and M. Latva-Aho, "Multi-operator spectrum sharing for small cell networks: A matching game perspective," *IEEE Trans. Wireless Commun.*, vol. 16, no. 6, pp. 3761–3774, Jun. 2017.

[41] B. Di, L. Song, and Y. Li, "Sub-channel assignment, power allocation, and user scheduling for non-orthogonal multiple access networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 11, pp. 7686–7698, Nov. 2016.

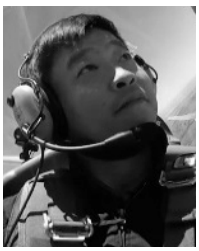
[42] T. Z. Oo, N. H. Tran, W. Saad, D. Niyato, Z. Han, and C. S. Hong, "Offloading in HetNet: A coordination of interference mitigation, user association, and resource allocation," *IEEE Trans. Mobile Comput.*, vol. 16, no. 8, pp. 2276–2291, Aug. 2017.

[43] L. Duan, L. Gao, and J. Huang, "Cooperative spectrum sharing: A contract-based approach," *IEEE Trans. Mobile Comput.*, vol. 13, no. 1, pp. 174–187, Jan. 2014.

TRA HUONG THI LE received the B.S. and M.S. degrees in electric and electronics engineering from the Ho Chi Minh City University of Technology, Ho Chi Minh City, Vietnam, in 2010 and 2012, respectively. She is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, Kyung Hee University. Her research interests include resource management and game theory.

NGUYEN H. TRAN (S'10–M'11–SM'18) received the B.S. degree in electrical and computer engineering from the Ho Chi Minh City University of Technology, in 2005, and the Ph.D. degree in electrical and computer engineering from Kyung Hee University, in 2011. He was an Assistant Professor with the Department of Computer Science and Engineering, Kyung Hee University, from 2012 to 2017. Since 2018, he has been with the School of Information Technologies, The University of Sydney, where he is currently a Senior Lecturer. His research interests include applying analytic techniques of optimization, game theory, and machine learning to cutting-edge applications, such as cloud and mobile-edge computing, datacenters, resource allocation for 5G networks, and the Internet of Things. He received the Best KHU Thesis Award in Engineering, in 2011, and several best paper awards, including the IEEE ICC 2016, the APNOMS 2016, and the IEEE ICCS 2016. He received the Korea NRF Funding for Basic Science and Research, from 2016 to 2023. He has been the Editor of the IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, since 2016.

PHUONG LUU VO received the B.Eng. and M.Eng. degrees in electrical and electronics engineering from the Ho Chi Minh City University of Technology, Vietnam, in 1998 and 2002, respectively, and the Ph.D. degree from Kyung Hee University, South Korea, in 2014. She is currently a Lecturer with the School of Computer Science and Engineering, International University-Vietnam National University Ho Chi Minh City. Her research interest includes applying optimization theory, control theory, and game theory to allocate the resources in communication networks.



ZHU HAN (S'01–M'04–SM'09–F'14) received the B.S. degree in electronic engineering from Tsinghua University, in 1997, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland, College Park, in 1999 and 2003, respectively. From 2000 to 2002, he was a Research and Development Engineer with JDSU, Germantown, MD, USA. From 2003 to 2006, he was a Research Associate with the University of Maryland. From 2006 to 2008, he was an Assistant Professor with Boise State University, ID, USA. He is currently a John and Rebecca Moores Professor with the Electrical and Computer Engineering Department, University of Houston, TX, USA, and also with the Computer Science Department, University of Houston. His research interests include wireless resource allocation and management, wireless communications and networking, game theory, big data analysis, security, and smart grid. He is currently an IEEE Communications Society Distinguished Lecturer. He was a recipient of the NSF Career Award, in 2010. He received the Fred W. Ellersick Prize from the IEEE Communication Society, in 2011, the EURASIP Best Paper Award for the *Journal on Advances in Signal Processing*, in 2015, the IEEE Leonard G. Abraham Prize in the field of communications systems (Best Paper Award in IEEE JSAC), in 2016, and several best paper awards in IEEE conferences.

MEHDI BENNIS (S'07–A'08–SM'15) received the M.Sc. degree in electrical engineering jointly from the EPFL, Lausanne, Switzerland, and the Eurecom Institute, France, in 2002, and the Ph.D. degree in spectrum sharing for future mobile cellular systems, in 2009. From 2002 to 2004, he was a Research Engineer with IMRA Europe, investigating adaptive equalization algorithms for mobile digital TV. In 2004, he joined the Centre for Wireless Communications, University of Oulu, Oulu, Finland, as a Research Scientist. In 2008, he was a Visiting Researcher with the Alcatel-Lucent Chair on Flexible Radio, SUPELEC. He is currently an Adjunct Professor with the University of Oulu. He has co-authored one book and published more than 100 research papers in international conferences, journals, and book chapters. His research interests include radio resource management, heterogeneous networks, game theory, and machine learning in 5G networks and beyond. He received the prestigious 2015 Fred W. Ellersick Prize from the IEEE Communications Society. He serves as an Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS.

CHOONG SEON HONG received the B.S. and M.S. degrees in electronic engineering from Kyung Hee University, Seoul, South Korea, in 1983 and 1985, respectively, and the Ph.D. degree from Keio University, Japan, in 1997. In 1988, he joined KT as a Member of Technical Staff, where he was involved in broadband networks. In 1993, he joined Keio University. He was with the Telecommunications Network Laboratory, KT, as a Senior Member of Technical Staff and as the Director of the Networking Research Team, until 1999. Since 1999, he has been a Professor with the Department of Computer Engineering, Kyung Hee University. His research interests include future Internet, ad hoc networks, network management, and network security. He is a member of the ACM, the IEICE, the IPSJ, the KIISE, the KICS, the KIPS, and the OSIA. He has served as the General Chair, as the TPC Chair/Member, or as an Organizing Committee Member for international conferences, such as NOMS, IM, APNOMS, E2EMON, CCNC, ADSN, ICPP, DIM, WISA, BcN, TINA, SAINT, and ICOIN. He is currently an Associate Editor of the IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT, the *International Journal of Network Management*, and the IEEE JOURNAL OF COMMUNICATIONS AND NETWORKS, and an Associate Technical Editor of *IEEE Communications Magazine*.