Online Resource Allocation, Content Placement and Request Routing for Cost-efficient Edge Caching in Cloud Radio Access Networks

Lingjun Pu, Lei Jiao, Xu Chen, *Member, IEEE,* Lin Wang, Qinyi Xie, and Jingdong Xu.

***Abstract*—In this paper, we advocate edge caching in Cloud Radio Access Networks (C-RAN) to facilitate the ever-increasing mobile multimedia services. In our framework, central offices will cooperatively allocate cloud resources to cache popular contents and satisfy user requests for those contents, so as to minimize the system costs in terms of storage, VM reconfiguration, content access latency, and content migration. However, this joint resource allocation, content placement and request routing is nontrivial, since it needs to be continuously adjusted to accommodate system dynamics, such as user movement and content slashdot effect, while taking into account the time-correlated adjustment costs for VM reconfiguration and content migration. To this end, we build a comprehensive model to capture the key components of edge caching in C-RAN, and formulate a joint optimization problem, aiming at minimizing the system costs over time, and meanwhile satisfying the time-varying user requests and respecting various practical constraints (e.g., storage and bandwidth). Then, we propose a novel online approximation algorithm by resorting to the regularization, rounding and decomposition technique, which can be proved to have a parameterized competitive ratio with a polynomial running time. Extensive trace-driven simulations corroborate the efficiency, flexibility and lightweight of our proposed online algorithm, for instance it achieves an empirical competitive ratio around 2 – 4, gains over 30% improvement compared with many state-of-the-art algorithms in various system settings.**

***Index Terms*—C-RAN, Edge Caching, Content Placement, Request Routing, Resource Allocation, Approximation Algorithm.**

I. Introduction

With the proliferation of mobile devices and the prosperity of Over-The-Top content providers, recent years have witnessed a dramatic increase in mobile multimedia services. The latest Cisco VNI report predicts that mobile multimedia traffic will grow at a CAGR of 54% between 2016 and 2021, and will account for 80% of overall mobile data traffic in 2021 [1]. This big and growing mobile multimedia traffic brings increasing pressure on radio access networks and backhaul transmission networks, and becomes the primary concern of mobile network operators, which motivates the innovations in the operations of future cellular networks such as pursuing novel network architectures and advanced content delivery technologies.

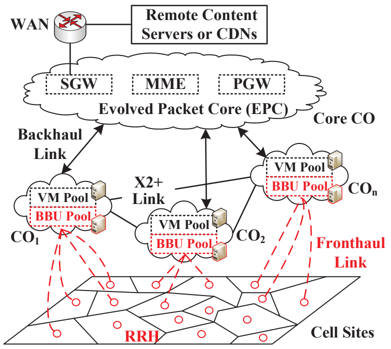


Fig. 1: A typical C-RAN scenario (CO: abbreviation of central office).

Recently, Cloud Radio Access Network (C-RAN) is proposed as a novel and promising architecture for future cellular networks by combining RAN with cloud computing [2]. In general, a typical C-RAN as shown in Fig. 1 consists of lightweight Radio Remote Heads (RRHs) deployed at cell sites that provide basic signal transmission and reception functionalities, virtualized BaseBand Unit (BBU) pools hosted in central offices (e.g., edge clouds or cloudlets) that conduct the baseband processing, and high-bandwidth, low-latency fronthaul links (X2+ links) connecting RRHs to BBU pools (BBU pools to BBU pools). Intuitively, as the BBUs from nearby cell sites are co-located in one or several close BBU pools, they can achieve low-latency interaction to increase spectral efficiency and facilitate signaling control. However, the ever-increasing mobile multimedia services still require high-capacity backhaul links for retrieving contents from remote content servers or CDNs, which implies that reforming network architectures solely is insufficient and has to be accompanied by innovations in content delivery.

Edge caching in cellular networks [3], [4] is deemed as the most effective content delivery solution to cope with the

Manuscript received December 10, 2017; revised April 08 and May 20, 2018; accepted May 26, 2018. This work was supported in part by the National Key Research and Development Program of China under grant No.2017YFB1001703; the National Natural Science Foundation of China (No.61702287, No.U1711265); the Fundamental Research Funds for the Central Universities (No.7933, No.17lgjc40); the Program for Guangdong Introducing Innovative and Enterpreneurial Teams (No.2017ZT07X355), and the Open Project Fund of Guangdong Key Laboratory of Big Data Analysis and Processing (No.2017003). *(Corresponding author: Lei Jiao and Xu Chen.)*

Lingjun Pu is with the College of Computer and Control Engineering, Nankai University, Tianjin 300071, China, and also with the Guangdong Key Laboratory of Big Data Analysis and Processing, Sun Yat-sen University, Guangzhou 510006, China. (e-mail: pulj@nankai.edu.cn).

Lei Jiao is with the Department of Computer and Information Science, University of Oregon, Eugene 97403, USA. (e-mail: jiao@cs.uoregon.edu).

Xu Chen is with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China. (e-mail: chenxu35@mail.sysu.edu.cn).

Lin Wang is with the Department of Computer Science, TU Darmstadt, Darmstadt 64289, Germany. (e-mail: wang@tk.tu-darmstadt.de).

Qinyi Xie and Jingdong Xu are with the College of Computer and Control Engineering, Nankai University, Tianjin 300071, China. (e-mail: {xieqy, xujd}@nankai.edu.cn).

multimedia traffic over cellular networks. Specifically, popular contents are cached in the RAN facilities (such as routers and base stations), and user requests for those contents can be fulfilled by local caches in the RAN without duplicate transmissions from content servers or CDNs over the Internet, which not only greatly eliminates redundant traffic but also efficiently improves user QoS (e.g., content access latency). In the last five years, many researchers pay great attention to caching in heterogenous base stations (i.e., Femtocaching) such as [5]–[8]. However, Femtocaching requires future base stations to be armed with additional storage units, which will increase network operators' capital expenditures, especially in the future densification era. Also, it is not scalable to adjust the storage capacities of base stations after their deployment. Moreover, the limited coverage of base stations degrades the caching performance. For example, a typical CDN cache normally receives 50 requests/content/day, while a base station cache may be as low as 0.1 requests/content/day [4]. Therefore, seeking for novel edge caching frameworks in cellular networks is still meaningful and significant.

Realizing the great potentials and open issues of these two techniques, we advocate edge caching in C-RAN to facilitate the ever-increasing mobile multimedia services. That is, central offices in C-RAN will cooperatively allocate cloud resources to cache popular contents and satisfy user requests for those contents in their service areas. The advantages of this framework are as follows. First, without modifying RRHs, central offices can provide a storage-based VM pool for content caching, which is easy to maintain and scalable. Second, besides its functionalities (e.g., MME), evolved packet core can interact with central offices to obtain the global system information, which helps to make holistic and efficient caching decisions. Third, each central office serves a group of cell sites, and this appreciable service coverage facilitates content caching taking effect. In order to reap its profound benefits, we require to solve a critical issue, the **joint resource allocation** (i.e., the decision about the allocated VM amount in each central office), **content placement** (i.e., the decision about the content availability in each central office) and **request routing** (i.e., the decision about which central offices the missing user requests in a central office should be redirected to), which involves multiple challenges:

(1) It is not a one-shot operation, but needs to be continuously adjusted to accommodate system dynamics such as user movements and newly generated content requests, which incurs the time-correlated adjustment costs to capture VM reconfiguration (e.g., the switching cost of VMs on/off [9], [10]) and content migration (e.g., the bandwidth cost of content downloading from remote content servers or other central offices [11], [12]) in each central office.

(2) It needs to operate on the fly. The main reasons are twofold. First, there are no well-established methods to accurately estimate how each user will move and what contents he will request over time [13]. Second, popular multimedia contents often come about the flash crowd phenomenon (a.k.a the slashdot effect), that is, they appear, then become increasingly popular, and gradually become unpopular again [3], [4], which leads to the time-varying user requests for them.

(3) It should be efficient, flexible and lightweight. Many previous works (e.g., [5]–[8], [11], [12]) indicate that the joint content placement and request routing belongs to the mixed-integer programming, which is NP-hard, and it is even more complicated when taking resource allocation as well as the aforementioned online and time correlation features into consideration. Therefore, a good online algorithm with a provable competitive ratio and polynomial running time which is also flexible to various system settings is highly desirable.

To address these challenges, we build a comprehensive model which captures the key components of edge caching in C-RAN and the holistic system costs. In this context, we formulate a joint resource allocation, content placement and request routing problem, aiming at minimizing the system costs over time, satisfying the time-varying user requests and respecting various practical constraints such as the limited storage capacity of each central office (Section III).

We design a novel online approximation algorithm by resorting to the regularization, rounding and decomposition techniques. Briefly, we transform the time-correlated adjustment costs in the original problem into carefully-designed logarithmic forms, relax the integer control variables to real ones, and decouple the transformed problem into a series of time-independent convex subproblems which can be efficiently solved with the previous and current system information (i.e., regularization). Taking the relaxed integer control variables into account, we propose a randomized dependent rounding algorithm, which produces a feasible integral solution for the outermost covering variables, according to the derived fractional solution of the subproblem at each time frame (i.e., rounding). We bring those rounded variables back to the original problem and adopt the dual-decomposition method to generate the optimal results for the remaining control variables (i.e., decomposition). Note that, the above three procedures operate in polynomial time and they assume no priori knowledge of user movement, content request preference and content popularity. (Section IV).

We prove that the proposed online algorithm can achieve a parameterized competitive ratio (Section V). Extensive trace-driven simulations validate the superior performance of our proposed algorithm. For example, it achieves an empirical competitive ratio around $2-4$, gains over 30% performance compared with many state-of-the-art algorithms, and consumes less than 12s in various system settings (Section VI).

II. Related Work

Due to the great potential of edge caching in cellular networks, the joint optimization problem of content placement and request routing has received great attention in recent years. In the beginning, numerous research efforts consider the joint optimization problem to minimize content access latency or maximize the amount of user requests served by the caches. Dehghan *et al.* [5] investigate two variants of the problem in terms of whether content access latency is dependent or independent of user request rate, and they formulate both of them as the maximization of a submodular function subject to matroid constraints, which can be efficiently solved by greedy algorithms. Poularakis *et al.* [14] define a specific objective to capture the access latency of layered videos, formulate the joint optimization problem as a multiple-choice knapsack problem, and design a novel approximation algorithm within a 2 factor from the optimum. They also consider to model user movements via random walks on a Markov chain, and design a distributed caching and routing paradigm with user mobility prediction and network coding [15]. Jiang *et al.* [7] leverage hierarchical primal-dual decomposition method to decouple the cooperative caching problem into two-level optimization problems, which are solved by using the subgradient method. Li *et al.* [8] provide an approximate decomposition method which decouples the cooperative caching problem into some subproblems that focus on the caching cooperation at different tiers, and they design greedy heuristic methods to respectively solve the subproblems. Besides, many other researchers also consider the similar objective with different scenarios or settings. Nevertheless, most of them either assume content popularity following the Zipf distribution (e.g., [8]) or assume user movement following some well-known models (e.g., [15]). More importantly, their models do not involve the content storage cost and the system adjustment costs considered in this paper, and therefore applying their proposed algorithms to our problem may not achieve the same performance.

A few researchers incorporate the storage cost into the joint optimization problem [6], [16]–[18]. Gharaibeh *et al.* [6] provide an approximation algorithm that does not require any knowledge of user movement and content popularity, while their model does not capture the limited storage and bandwidth capacity of base stations. Shukla *et al.* [16], [17] introduce the node storage constraint and develop efficient approximation algorithms, while they focus on the proactive caching which requires the knowledge of content popularity. Hou *et al.* [18] design an approximation algorithm without capturing the cooperation between edge clouds. Besides, these works also do not involve the system adjustment costs.

The time-correlated adjustment costs cannot be neglected in online systems and have got widespread traction in many domains such as content delivery [11], [19], [20] and resource allocation [9], [10], [12], [21] in hybrid clouds, BS sleeping in cellular networks [22], [23], and energy generation in microgrids [24], [25]. The authors in [19], [23] consider the workload queueing model, and invoke Lyapunov optimization technique to discuss the time-average performance. The authors in [12], [20] derive the one-shot (optimal) solution, and exploit online prediction and Δt-step look-ahead technique to adjust each one-shot solution towards offline optimum, which however cannot achieve a competitive ratio. Our work differs from them, since our purpose is to design an efficient approximation algorithm for the joint resource allocation, content placement and request routing problem in the C-RAN based edge caching framework, without introducing user request queueing model due to the frequent user movement and the stringent delay requirement of multimedia contents. Although the work [20] and our previous work [10], [21] also exploit the regularization technique [26] to design approximation algorithms, similar to the other works, they do not capture the unique “triangular covering” feature and the boxing constraints of the control variables in our problem, leading to a quite different algorithm design and theoretical analysis.

There are also some recent researches touching on edge caching in C-RAN such as [27], [28], while they only focus on the minimization of content access latency and assume the content popularity is static and known in advance. **To the best of our knowledge, our work is the first to fully take the holistic system costs in terms of storage, VM reconfiguration, content access latency and content migration, as well as the storage and bandwidth constraints into the joint resource allocation, content placement and request routing problem in the C-RAN based edge caching framework, and our key contribution is to develop an online efficient approximation algorithm to solve this novel joint optimization problem with a provable competitive ratio.** We hope our proposal will provide some guidelines and thoughts to boost both edge caching and C-RAN research.

III. System Model and Problem Formulation

We consider a general C-RAN scenario as shown in Fig. 1 which consists of a set $\mathcal{N}=\{1,2,\ldots,N\}$ of central offices, and each central office serves a group of cell sites via the high-bandwidth, low-latency fronthaul links (e.g., passive optical networks). We assume these central offices are cooperative to build an edge caching framework, aiming at attracting multimedia content providers to participate in the edge caching service, in which they will connect with each other via the emerging high-bandwidth X2+ links [2] to support content sharing (i.e., request routing), and a shared core central office will make a cost-efficient edge caching decision on the joint resource allocation, content placement and request routing in a holistic manner[1]. In addition, we envision the joined multimedia content providers has a set of $\mathcal{M}=\{1,2,\ldots,M\}$ contents which will be requested by mobile users over time (M can be an arbitrarily large number), and assume these contents are of equal size S. Note that, this assumption is justified in real systems which break contents into equal size chunks and has been mentioned in many previous works (e.g., [5], [7], [16], [17], [28]). For simplicity, we define S as the unit size (i.e., $S=1$). Besides, we consider this edge caching framework operates in a discrete-time manner, and each time frame $t\in\mathcal{T}=\{1,2,\ldots,T\}$ has a moderate time length (e.g., dozens of minutes), determined by the periodicity in user request patterns for multimedia contents.

User Requests: We consider user requests at the granularity of central offices. That is, we assume each central office i will receive an accumulated number of user requests n_{it}^{c} for content c from its served cell sites at a time frame t. The value of n_{it}^{c} can vary over time, due to new user arrival, existing user departure and content slashdot effect, which can be only known or predicted at the beginning of each time frame. Note that this setting enables our framework not to care about the network management and resources scheduling (e.g., RRH sleeping and user association) in the BBU pool, leading to the compatibility and flexibility in practice.

[1] Note that, this assumption can be easily achieved by one or multiple cooperative mobile network operators with the deployed center offices across administrative regions in a city or across cities in a province [12], [28].

Resource Allocation: We consider each central office j will offer a specific setting of VM instance $<D_j, B_j>$ for the edge caching service, determined by its associated mobile network operators, where D_j indicates the maximum number of contents stored by one VM, and B_j indicates the maximum number of user connections supported (or the maximum bandwidth provided) by one VM. To capture the limited resources of each central office (e.g., edge cloud or cloudlet), we denote by C_j the storage capacity of central office j, which refers to the maximum number of active VMs[2]. Then, we introduce an integer control variable z_{jt} to indicate the allocated VM amount in central office j at time frame t. Intuitively, we have $z_{jt} \in \{0, 1, \ldots, C_j\}$. Besides, we denote the unit storage cost by c_{jt}, to capture the rent price and the maintenance expense incurred per active VM, which can be dynamically adjusted by mobile network operators in terms of many practical factors (e.g., the available storage capacity at different time).

Service Model: With the allocated VM resources, we consider each central office will serve its received user requests in two ways: on the one hand, it can retrieve some "uncached" contents from remote content servers or other central offices, and fulfill the user requests for those contents by itself; on the other hand, it can redirect some user requests to other central offices which have cached the requested contents, and let those central offices fulfill the user requests. We capture both of them[3] with the content placement and request routing.

Content Placement and Request Routing: As for the content placement, we introduce a binary control variable y_{jt}^c to indicate whether content c is cached in central office j at time frame t. As the number of cached contents cannot exceed the allocated storage capacity in each central office, we have the storage constraint: $\sum_{c=1}^{M} y_{jt}^c \le D_j z_{jt}, \forall j \in \mathcal{N}$. Note that the cached contents in each central office can be discarded at any time frame, in terms of many practical issues (e.g., low user request amount and/or high storage cost).

As for the request routing, we introduce a control variable x_{ijt}^c to indicate the percentage of user requests for content c in central office i served by central office j at time frame t. Since each content can be retrieved from a central office only if it has been cached in there, we have the following precedence constraint: $x_{ijt}^c \le y_{jt}^c, \forall i, j \in \mathcal{N}, \forall c \in \mathcal{M}$. In addition, we introduce the user connection constraint[4]: $\sum_{i=1}^{N} \sum_{c=1}^{M} x_{ijt}^c n_{it}^c \le B_j z_{jt}, \forall j \in \mathcal{N}$, which captures the total number of user requests served by a central office cannot exceed its allocated user connection capacity (i.e., the application layer constraint as discussed in [16]). Besides, we introduce the service integrity constraint: $\sum_{j=1}^{N} x_{ijt}^c \ge \mathbb{1}_{\{n_{it}^c \ge 1\}}, \forall i \in \mathcal{N}, \forall c \in \mathcal{M}$, which indicates that the user requests for any content should be satisfied by the edge caching service entirely.

We denote the unit request routing cost by d_{ij}, which captures the delay of transferring one content between central office i, j plus that between central office i and its associated users, the monetary price for using the bandwidth of X2+ links en route, and some costs associated with the user-central office session in the application layer. We should emphasize that, this model does not care about the specific models of X2+, fronthaul and cellular links, since the BBU pool in each central office can obtain or estimate various kinds of network information (e.g., transmission delay), and pass them to the core central office to facilitate the edge caching service.

Holistic System Costs: Our model focuses on four kinds of system costs. The first two are static costs associated with every independent time frame: the storage cost which captures the allocated VM amount in central offices (i.e., $\sum_{j=1}^{N} c_{jt} z_{jt}$); the request routing cost which captures the content access latency of user requests (i.e., $\sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{c=1}^{M} d_{ij} x_{ijt}^c n_{it}^c$). The rest two are dynamic costs related to each pair of consecutive time frames: the VM reconfiguration cost which captures the changes of allocated VM amount in central offices (i.e., $\sum_{j=1}^{N} s_j [z_{jt} - z_{jt-1}]^+$, where s_j represents the cost of increasing one VM instance (e.g., booting and service deployment) in central office j, and $[x]^+ \triangleq \max\{x, 0\}$); the content migration cost which captures the number of new downloading contents in central offices (i.e., $\sum_{j=1}^{N} \sum_{c=1}^{M} b_j [y_{jt}^c - y_{jt-1}^c]^+$, where b_j is the average delay and bandwidth cost of downloading one content from a specific remote content server which permanently stores all the content copies in the system[5] [11]. Note that, all the cost parameters can be specified by mobile network operators or estimated by central offices.

Problem Formulation: Our main purpose is to develop a cost-efficient edge caching policy in the aforementioned framework, to attract multimedia content providers to take part in the edge caching service. To this end, we formulate a joint resource allocation, content placement and request routing problem, aiming at minimizing the holistic system costs over time, and meanwhile satisfying the time-varying user requests and respecting various system constraints as discussed above. It can be mathematically formulated as follows:

$$
\begin{aligned}
\min \quad \mathbf{P}_1 &= \sum_t \sum_j c_{jt} z_{jt} + \sum_t \sum_i \sum_j \sum_c d_{ij} x_{ijt}^c n_{it}^c \\
&\quad + \sum_t \sum_j s_j [z_{jt} - z_{jt-1}]^+ + \sum_t \sum_j \sum_c b_j [y_{jt}^c - y_{jt-1}^c]^+ \\
\text{s.t.} \quad & \textstyle\sum_j x_{ijt}^c \ge \mathbb{1}_{\{n_{it}^c \ge 1\}}, \qquad \forall i,\ \forall c,\ \forall t, && \text{(1a)} \\
& y_{jt}^c \ge x_{ijt}^c, \qquad \forall i,\ \forall j,\ \forall c,\ \forall t, && \text{(1b)} \\
& B_j z_{jt} \ge \textstyle\sum_i \sum_c n_{it}^c x_{ijt}^c, \qquad \forall j,\ \forall t, && \text{(1c)} \\
& D_j z_{jt} \ge \textstyle\sum_c y_{jt}^c, \qquad \forall j,\ \forall t. && \text{(1d)} \\
\text{var} \quad & x_{ijt}^c \in [0, 1],\ y_{jt}^c \in \{0, 1\},\ z_{jt} \in \{0, \ldots, C_j\}. && \text{(1e)}
\end{aligned}
$$

[2] We can also consider the bandwidth capacity of each central office in the model, without affecting the algorithm design. Also, our framework supports the time-varying storage capacity, in the context of the resource competition among edge caching, BBU pool and edge computing in a central office.

[3] There may exist other service model such as hierarchical caching model which takes the core central office and/or remote content servers into account [5], [28]. Compared with them, our "plain" caching model is proposed from the operator's perspective, which is easy to implement and manage in practice, and we will integrate their models with ours in the future work.

[4] Note that, it can also be viewed as the bandwidth constraint if B_j represents the maximum bandwidth provided by one VM. That is, $\sum_{i=1}^{N} \sum_{c=1}^{M} x_{ijt}^c n_{it}^c S \le B_j z_{jt}, \forall j \in \mathcal{N}$, where S is the content size as mentioned above.

[5] Our framework can incorporate other predesigned content migration strategies. For example, we can allow each central office to download the content from the nearest central office with the required one. In this case, we need to substitute b_j with $b_j^c(t)$. Note that, although $b_j^c(t)$ may be time-varying, it is an input parameter in our formulated problem, and accordingly does not affect our algorithm design.

IV. Online Algorithm Design

Algorithmic Challenges: The main challenges of this joint optimization problem stem from three aspects: (1) it is not easy to make a good joint decision on the fly, since the problem external inputs (e.g., user request amount and unit storage cost[6]) arrive online and are difficult to predict in advance. Meanwhile, the values of the problem objective cannot be horizontally decoupled over time due to two time-correlated adjustment costs. (2) it is also hard to make integral decisions on the resource allocation and content placement in an online manner, since this problem belongs to mixed-integer programming, which is NP-Hard and difficult to tackle even in the offline case. (3) the inherent structure of this problem such as the unique "triangular covering" feature (i.e., z covers x, y while y covers x) and the boxing constraints of control variables further complicates the optimal algorithm design.

Uniqueness: Note that some recent works also discuss the scheduling difficulty caused by the time-correlated adjustment costs [9]–[12], [19], [20], [22]–[25]. Nevertheless, besides some of them do not consider the algorithm design in the online approximation perspective (e.g., offline [22], heuristics [11], [12] and time-average approximation [19], [23]), their inherent problem structures such as no integer control variables [9], [10], only involving precedence constraints [20], [25] or introducing ramp constraints [24] are greatly different from ours, leading to the infeasibility of their proposed algorithms for our problem. Therefore, it is of great significance to design a new algorithm for such a novel and challenging problem.

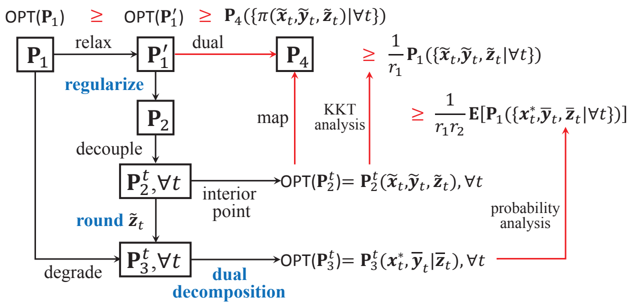

OPT($\mathbf{P}_1$) ≥ OPT($\mathbf{P}_1'$) ≥ $\mathbf{P}_4(\{\pi(\tilde{\boldsymbol{x}}_t,\tilde{\boldsymbol{y}}_t,\tilde{\mathbf{z}}_t)|\forall t\})$
$\mathbf{P}_1$
relax
$\mathbf{P}_1'$
dual
$\mathbf{P}_4$
$\geq \frac{1}{r_1}\mathbf{P}_1(\{\tilde{\boldsymbol{x}}_t,\tilde{\boldsymbol{y}}_t,\tilde{\mathbf{z}}_t|\forall t\})$
regularize
$\mathbf{P}_2$
map
KKT analysis
$\geq \frac{1}{r_1 r_2}\mathbf{E}[\mathbf{P}_1(\{\boldsymbol{x}_t^*,\bar{\boldsymbol{y}}_t,\bar{\mathbf{z}}_t|\forall t\})]$
decouple
$\mathbf{P}_2^t,\forall t$
interior point
OPT($\mathbf{P}_2^t$)= $\mathbf{P}_2^t(\tilde{\boldsymbol{x}}_t,\tilde{\boldsymbol{y}}_t,\tilde{\mathbf{z}}_t),\forall t$
round $\tilde{\mathbf{z}}_t$
probability analysis
$\mathbf{P}_3^t,\forall t$
degrade
dual decomposition
OPT($\mathbf{P}_3^t$)= $\mathbf{P}_3^t(\boldsymbol{x}_t^*,\bar{\boldsymbol{y}}_t|\bar{\mathbf{z}}_t),\forall t$

Fig. 2: Basic idea of our online algorithm (black arrows) and main route of our performance analysis (red arrows).

Basic Idea: In this section, we develop a novel online approximation algorithm, resorting to the *regularization*, *rounding* and *decomposition* technique, to conquer the aforementioned challenges. Our basic idea is shown in Fig. 2, which consists of the following steps:

- **Step** 1: We relax the integer control variables in (1e), and allow them to take real values. Then, we transform the relaxed problem $\mathbf{P}_1'$ into a more tractable problem $\mathbf{P}_2$ via the regularization technique [26], i.e., by replacing the time-correlated adjustment costs in the objective with carefully-designed logarithmic forms. This transformed problem can be easily decoupled into a series of time-independent convex subproblems $\{\mathbf{P}_2^t, \forall t\}$, each of which can be efficiently solved with the previous and current system information.
- **Step** 2: Taking the integral constraints and the "triangular covering" feature of control variables into account, we develop a novel randomized dependent rounding algorithm, inspired from the pipage rounding technique [29], [30], which aims to round the fractional solution of resource allocation (i.e., the outermost covering variables) into integral one, while maintaining a feasible content placement and request routing solution to the original problem at each time frame.
- **Step** 3: We bring the derived integral solution of resource allocation back to the original problem which consequently degrades to a joint content placement and request routing problem $\mathbf{P}_3^t$ at each time frame. Then, we propose an efficient algorithm to obtain the optimal solution of this problem via the dual-decomposition technique. Finally, the derived optimal solution of $\mathbf{P}_3^t$ together with the integral solution of resource allocation constitutes the entire solution of the original problem at each time frame.

In the following, we will concretely discuss our online algorithm, consisting of an Online Regularization-based Fractional Algorithm (ORFA), a Randomized Dependent Rounding Algorithm (RDRA) and a Dual Decomposition-based Optimal Algorithm (DDOA). Before that, we briefly introduce some notations which are used throughout the rest of paper. $\mathbf{z}_t$ is the shorthand for the set $\{z_{jt}, \forall j\}$, so is $\mathbf{x}_t$ for x_{ijt}^c, $\mathbf{y}_t$ for y_{jt}^c and others (e.g., $\mathbf{c}_t$ for c_{jt}). $(\tilde{\mathbf{x}}_t, \tilde{\mathbf{y}}_t, \tilde{\mathbf{z}}_t)$ refers to the fractional solution produced by the ORFA, $\bar{\mathbf{z}}_t$ refers to the integral solution produced by the RDRA, and $(\mathbf{x}_t^*, \bar{\mathbf{y}}_t)$ refers to the optimal solution produced by the DDOA.

A. Online Regularization-based Fractional Algorithm

Motivation: We introduce the regularization technique [26] to address the non-convex challenge, caused by the time-correlated adjustment costs. Taking the VM reconfiguration cost as an example, the main idea of this technique is to substitute the non-convex term $\sum_j s_j[z_{jt} - z_{jt-1}]^+$ with a regularized convex function at each time frame, which can facilitate the design of online approximation algorithms. Since this non-convex term can be approximately interpreted as the L1-distance and the relative entropy is known as an efficient alternative regularizer to the L1-distance in online learning problems [26], [31], we exploit the unnormalized relative entropy $\sum_j \frac{s_j}{\sigma_j}\left((z_{jt}+\varepsilon)\ln\frac{z_{jt}+\varepsilon}{z_{jt-1}+\varepsilon} - z_{jt}\right)$ to substitute the VM reconfiguration cost in our problem[7]. As a result, we can

[6] In this paper, we only consider the dynamic unit storage cost in our algorithm design and performance analysis, while our framework can support a range of time-varying parameters such as C_j, d_{ij} and b_j.

[7] Note that the original form of the unnormalized relative entropy should be $\sum_j \frac{s_j}{\sigma_j}\left((z_{jt}+\varepsilon)\ln\frac{z_{jt}+\varepsilon}{z_{jt-1}+\varepsilon} + z_{jt-1} - z_{jt}\right)$ [26], [31]. We omit the term z_{jt-1} here, in order to simplify the performance analysis (i.e., the theoretical proof) of our proposed online algorithm. In other words, our online algorithm can achieve the same performance when taking the original form into account. We should emphasize that, this "L1-distance and relative entropy" substitution should not be the only choice for our problem, and we leave further explorations on this issue in the future work.

generate the relaxed and regularized problem $\mathbf{P}_2$ from $\mathbf{P}_1$:

$$\begin{aligned} \min \quad \mathbf{P}_2 &= \sum_t \mathbf{P}_2^t = \sum_t\sum_j c_{jt}z_{jt} + \sum_t\sum_i\sum_j\sum_c d_{ij}x_{ijt}^c n_{it}^c \\ &+ \sum_t\sum_j \frac{s_j}{\sigma_j}\left((z_{jt}+\varepsilon)\ln\frac{z_{jt}+\varepsilon}{z_{jt-1}+\varepsilon} - z_{jt}\right)\\ &+ \sum_t\sum_j\sum_c \frac{b_j}{\eta}\left((y_{jt}^c+\varepsilon')\ln\frac{y_{jt}^c+\varepsilon'}{y_{jt-1}^c+\varepsilon'} - y_{jt}^c\right)\\ \text{s.t.} \quad & (1\text{a})-(1\text{d}),\\ \text{var} \quad & x_{ijt}^c\in[0,1],\ \ y_{jt}^c\in[0,1],\ \ z_{jt}\in[0,C_j]. \end{aligned}$$

Note that, we deliberately set the parameter $\sigma_j = \ln(1+\frac{C_j}{\varepsilon})$ and $\eta = \ln(1+\frac{1}{\varepsilon'})$, which will be used in the performance analysis, while we can set the parameter ε and ε' to arbitrary positive values, since they are introduced to guarantee the non-zero denominator in the $\ln$ operator.

Algorithm Design: In terms of the above formulation, we find that the regularized problem $\mathbf{P}_2$ can be decoupled into a series of time-independent convex subproblems $\{\mathbf{P}_2^t, \forall t\}$. Since each subproblem $\mathbf{P}_2^t$ is a standard convex problem, we can invoke many mature algorithms such as interior point methods [26] to efficiently solve it[8]. In this context, we develop an Online Regularization-based Fractional Algorithm (ORFA) as show in Alg. 1, which generates the fractional solution $(\widetilde{\mathbf{x}}_t, \widetilde{\mathbf{y}}_t, \widetilde{\mathbf{z}}_t)$ greedily at each time frame by using the previous and current system information.

Algorithm 1: ORFA

```
Input: $\mathcal{N}$, $\mathcal{M}$, $\mathbf{D}$, $\mathbf{B}$, $\varepsilon$, $\varepsilon'$, $\widetilde{\mathbf{x}}_0 = \widetilde{\mathbf{y}}_0 = \widetilde{\mathbf{z}}_0 = 0$
Output: $(\widetilde{\mathbf{x}}_t, \widetilde{\mathbf{y}}_t, \widetilde{\mathbf{z}}_t)$, $\forall t \in \mathcal{T}$
while time frame $t \in \{1, 2, \ldots, T\}$ begins do
    Get $\boldsymbol{c}_t$, $\boldsymbol{d}$, $\boldsymbol{n}_t$, $\boldsymbol{s}$, $\boldsymbol{b}$, $\boldsymbol{\sigma}$, $\boldsymbol{\eta}$, $\widetilde{\mathbf{y}}_{t-1}$, $\widetilde{\mathbf{z}}_{t-1}$;
    Invoke interior point methods to solve the problem $\mathbf{P}_2^t$:
        min    $\sum_j c_{jt}z_{jt} + \sum_i\sum_j\sum_c d_{ij}x_{ijt}^c n_{it}^c$
               $+\sum_j \frac{s_j}{\sigma_j}\left((z_{jt}+\varepsilon)\ln\frac{z_{jt}+\varepsilon}{z_{jt-1}+\varepsilon} - z_{jt}\right)$
               $+\sum_j\sum_c \frac{b_j}{\eta}\left((y_{jt}^c+\varepsilon')\ln\frac{y_{jt}^c+\varepsilon'}{y_{jt-1}^c+\varepsilon'} - y_{jt}^c\right)$
        s.t.   $(1a)-(1d)$, without $\forall t \in \mathcal{T}$.
        var    $x_{ijt}^c \in [0,1]$, $y_{jt}^c \in [0,1]$, $z_{jt} \in [0, C_j]$.
    Return the near-optimal solution $(\widetilde{\mathbf{x}}_t, \widetilde{\mathbf{y}}_t, \widetilde{\mathbf{z}}_t)$;
```

We should emphasize that, the derived fractional solution $(\widetilde{\mathbf{x}}_t, \widetilde{\mathbf{y}}_t, \widetilde{\mathbf{z}}_t)$ is the base of the final solution to the original problem at each time frame, which will be used as an external input in the next randomized dependent rounding algorithm.

B. Randomized Dependent Rounding Algorithm

Motivation: Taking the integral constraints of control variables into account, we require to round the derived fractional $\widetilde{\mathbf{y}}_t$, $\widetilde{\mathbf{z}}_t$ to integral $\overline{\mathbf{y}}_t$, $\overline{\mathbf{z}}_t$, and meanwhile make sure the rounded ones are still feasible in terms of the constraint (1a) – (1d) at each time frame. According to the "triangular covering" feature of control variables, we first round the outermost covering variables[9] $\widetilde{\mathbf{z}}_t$ in this subsection. Intuitively, the independent rounding policy where each control variable is rounded up or down independently of others may not always generate a feasible rounded solution (e.g., all variables are rounded down), and the conservative rounding policy where all the control variables are rounded up may lead to excessive system costs. Therefore, we design a novel randomized dependent rounding algorithm, inspired from the pipage rounding technique [29], [30], and the key idea is to compensate the round-down control variables with the round-up ones, while ensuring that the rounded solution is feasible for the original problem.

Algorithm Design: First, we will select a special central office j^* from $\mathcal{N}$, and it may require to activate extra VMs to take in some "homeless" contents, due to the round-down fractional VM amounts in other central offices. Then, we introduce two sets $\mathcal{N}_{\mathbb{Z}t} \triangleq \{j \mid \widetilde{z}_{jt} \in \mathbb{Z}, j \neq j^*\}$ and $\mathcal{N}_{\mathbb{R}t} \triangleq \{j \mid \widetilde{z}_{jt} \in \mathbb{R}^+, j \neq j^*\}$, in terms of the fractional solution $\widetilde{\mathbf{z}}_t$. Intuitively, $\mathcal{N}_{\mathbb{Z}t} \cup \mathcal{N}_{\mathbb{R}t} \cup \{j^*\} = \mathcal{N}$ at each time frame. For each central office $j_k \in \mathcal{N}_{\mathbb{R}t}$, we provide a probability coefficient p_{j_k} and a weight coefficient ω_{j_k} associated with it. To facilitate such settings, we next need to break the "triangular covering" feature of control variables, otherwise we can hardly determine the weight coefficients and the special central office. To this end, our basic idea is to respectively consider the constraint (1c) and (1d) in two independent rounding cases, and then combine the independent results as the final rounding result. For a clear description without duplication, we mainly discuss the case with the constraint (1d) in the following analysis. In this case, we will choose the special central office j^* with the minimal storage cost per VM per content (i.e., $j^* = \arg\min \frac{c_{jt}}{D_j}, j \in \mathcal{N}$). Then, we define $p_{j_k} \triangleq \widetilde{z}_{j_k t} - \lfloor \widetilde{z}_{j_k t} \rfloor$ and $\omega_{j_k} \triangleq D_{j_k}$ for each $j_k \in \mathcal{N}_{\mathbb{R}t}$, and also define $p_{j^*} \triangleq \widetilde{z}_{j^* t} - \lfloor \widetilde{z}_{j^* t} \rfloor$ and $\omega_{j^*} \triangleq D_{j^*}$.

In this context, we develop a Randomized Dependent Rounding Algorithm (RDRA) as shown in Alg. 2, which generates an integral solution $\overline{\mathbf{z}}_t$ at each time frame. This algorithm runs a series of rounding iterations, in each of which we randomly select two elements j_1, j_2 from $\mathcal{N}_{\mathbb{R}t}$, and let the product of the probability and the weight of the round-down (round-up) element add to (deduct from) that of the other element (i.e., the key idea of our dependent rounding), which also leads the probability of one of them to 0 or 1, in terms of the coupled coefficient γ_1 and γ_2 (i.e., line $2-7$). Note that, the above processing at least decreases the number of elements in $\mathcal{N}_{\mathbb{R}t}$ by 1 in each iteration (i.e., line 8, 9). When $\mathcal{N}_{\mathbb{R}t}$ has only one element (i.e., at the last iteration), our algorithm directly rounds it up with its current probability, and adopt the special central office j^* to fill the "storage gap", in terms of the rounding result of that element (i.e., line 14 – 17). Note that, if $\mathcal{N}_{\mathbb{R}t}$ exactly has no residual element after

[8] In practice, we can exploit the primal-dual interior point method to produce the ϵ_{tol}-solution for our problem in polynomial time [32], [33]. Here, ϵ_{tol} refers to the tolerance of the duality gap which indicates the accuracy of the derived solution compared with the exact optimal solution. Note that since this method often has faster than linear convergence, it is common to choose a very small ϵ_{tol} (e.g., 10^{-8}), which enables the derived ϵ_{tol}-solution to be very close to the optimal solution [32], [33].

[9] Note that, if we first round the middle covering variables $\widetilde{\mathbf{y}}_t$, then we may not always obtain a feasible $\overline{\mathbf{z}}_t$, due to the boxing constraint of $\mathbf{z}_t$ (i.e., $z_{jt} \in \{0, 1, \ldots, C_j\}$) and the constraint (1d). More importantly, we cannot establish the relationship between $\overline{\mathbf{z}}_t$ and $\widetilde{\mathbf{z}}_t$ in this case, and therefore cannot derive a proper competitive ratio (i.e., no performance guarantee).

Algorithm 2: RDRA

```
   Input:  $\mathcal{N}_{\mathbb{Z}t}$, $j^*$, $\{p_{j_k}, \omega_{j_k}\}$ of each $j_k \in \mathcal{N}_{\mathbb{R}t}$
   Output: $\mathcal{N}_{\mathbb{Z}t}$
1  while $|\mathcal{N}_{\mathbb{R}t}| > 1$ do
2      Randomly select two elements $j_1$, $j_2$ from $\mathcal{N}_{\mathbb{R}t}$;
3      Introduce $\gamma_1 \triangleq \min\{1-p_{j_1}, \frac{\omega_{j_2}}{\omega_{j_1}} p_{j_2}\}$,
       $\gamma_2 \triangleq \min\{p_{j_1}, \frac{\omega_{j_2}}{\omega_{j_1}}(1-p_{j_2})\}$;
4      With the probability $\frac{\gamma_2}{\gamma_1+\gamma_2}$ set
5      $p_{j_1} = p_{j_1} + \gamma_1$,  $p_{j_2} = p_{j_2} - \frac{\omega_{j_1}}{\omega_{j_2}}\gamma_1$;
6      With the probability $\frac{\gamma_1}{\gamma_1+\gamma_2}$ set
7      $p_{j_1} = p_{j_1} - \gamma_2$,  $p_{j_2} = p_{j_2} + \frac{\omega_{j_1}}{\omega_{j_2}}\gamma_2$;
8      If $p_{j_1} \in \{0, 1\}$, then set $\bar{z}_{j_1 t} = p_{j_1} + \lfloor \widetilde{z}_{j_1 t} \rfloor$,
       $\mathcal{N}_{\mathbb{Z}t} = \mathcal{N}_{\mathbb{Z}t} \cup \{j_1\}$,  $\mathcal{N}_{\mathbb{R}t} = \mathcal{N}_{\mathbb{R}t} \setminus \{j_1\}$;
9      If $p_{j_2} \in \{0, 1\}$, then set $\bar{z}_{j_2 t} = p_{j_2} + \lfloor \widetilde{z}_{j_2 t} \rfloor$,
       $\mathcal{N}_{\mathbb{Z}t} = \mathcal{N}_{\mathbb{Z}t} \cup \{j_2\}$,  $\mathcal{N}_{\mathbb{R}t} = \mathcal{N}_{\mathbb{R}t} \setminus \{j_2\}$;
10 if $|\mathcal{N}_{\mathbb{R}t}| = 0$ then
11     Set $p_{j^*} = \lceil p_{j^*} \rceil$;
12 if $|\mathcal{N}_{\mathbb{R}t}| = 1$ then
13     Select the only element $j_1$ from $\mathcal{N}_{\mathbb{R}t}$;
14     With the probability $p_{j_1}$ set $p_{j_1} = 1$,
       $p_{j^*} = p_{j^*} - \frac{\omega_{j_1}}{\omega_{j^*}}(1-p_{j_1})$;
15     With the probability $1 - p_{j_1}$ set $p_{j_1} = 0$,
       $p_{j^*} = p_{j^*} + \frac{\omega_{j_1}}{\omega_{j^*}} p_{j_1}$;
16     Set $\bar{z}_{j_1 t} = p_{j_1} + \lfloor \widetilde{z}_{j_1 t} \rfloor$,  $p_{j^*} = \lceil p_{j^*} \rceil$,
       $\mathcal{N}_{\mathbb{Z}t} = \mathcal{N}_{\mathbb{Z}t} \cup \{j_1\}$,  $\mathcal{N}_{\mathbb{R}t} = \mathcal{N}_{\mathbb{R}t} \setminus \{j_1\}$;
17 Set $\bar{z}_{j^* t} = p_{j^*} + \lfloor \widetilde{z}_{j^* t} \rfloor$,  $\mathcal{N}_{\mathbb{Z}t} = \mathcal{N}_{\mathbb{Z}t} \cup \{j^*\}$;
```

the iterations, then we can simply round up the probability of the special central office (i.e., line 11).

We can substitute D_j with B_j to determinate the weight coefficients and the special central office, and invoke the same algorithm to work for the other case with the constraint (1c). If we leverage $\mathcal{N}^1_{\mathbb{Z}t} = \{\bar{z}^1_{jt}, j \in \mathcal{N}\}$ and $\mathcal{N}^2_{\mathbb{Z}t} = \{\bar{z}^2_{jt}, j \in \mathcal{N}\}$ to respectively indicate the rounded results in two independent cases, then we can merge them to obtain the final integral solution $\bar{\mathbf{z}}_t = \big\{ \max\{\bar{z}^1_{jt}, \bar{z}^2_{jt}\}, j \in \mathcal{N} \big\}$.

Remarks: We next show that there are two important properties achieved by the RDRA, which will be exploited in the performance analysis.

(1) *Marginal Distribution Property*. The probability p_{j_k} of each element $j_k \in \mathcal{N}_{\mathbb{R}t}$ satisfies $\Pr(X_{j_k} = 1) = p_{j_k}$, where X_{j_k} is a binary random variable indicating the rounded value of p_{j_k} produced by the RDRA.

(2) *Weight Conservation Property*. The rounded probability $\bar{p}_{j_k}$ and the corresponding weight ω_{j_k} of each element $j_k \in \mathcal{N}_{\mathbb{R}t}$ satisfies

$$\begin{aligned} \omega_{j^*} p_{j^*} + \textstyle\sum_{j_k \in \mathcal{N}_{\mathbb{R}t}} \omega_{j_k} p_{j_k} &\leq \omega_{j^*} \bar{p}_{j^*} + \textstyle\sum_{j_k \in \mathcal{N}_{\mathbb{R}t}} \omega_{j_k} \bar{p}_{j_k} \\ &\leq (1+\Psi)\omega_{j^*} + \omega_{j^*} p_{j^*} + \textstyle\sum_{j_k \in \mathcal{N}_{\mathbb{R}t}} \omega_{j_k} p_{j_k}, \end{aligned}$$

where $\Psi \triangleq \max\big\{ \lceil \frac{\omega_{j_k}}{\omega_{j^*}} \rceil, \forall j_k \in \mathcal{N}_{\mathbb{R}t} \big\}$. To avoid distracting attention from our algorithm design, we leave the detail proofs of these two properties in the online technical report [34].

Feasibility Analysis: The RDRA rounds the probability p_{j_k} of each element $j_k \in \mathcal{N}_{\mathbb{R}t}$ to either 0 or 1, which indicates the value of $\bar{z}_{j_k t}$ cannot exceed that of $\lfloor \widetilde{z}_{j_k t} \rfloor$ plus 1, and therefore we can declare that the integral solution $\{\bar{z}_{jt}, j \in \mathcal{N} \setminus j^*\}$ produced by the RDRA does not violate the boxing constraint (1e). However, as for the special central office j^*, it may not have sufficient storage resources to fill the "storage gap" caused by the round-down last element in $\mathcal{N}_{\mathbb{R}t}$ (i.e., line 15 – 17) at some time frames, that is $\bar{z}_{j^* t} \geq C_{j^*}$, which will violate the boxing constraint and invalidate our algorithm. To prevent this situation happening in practice, let us first consider the case with the constraint (1d). In this case, we can reserve Ψ VMs in j^* to facilitate rounding, and substitute C_{j^*} with the "shrinking" storage capacity $C'_{j^*} = C_{j^*} - \Psi$ in the original problem. Here, $\Psi \triangleq \max\{\lceil D_j / D_{j^*} \rceil\}$ in the considered case as defined above. Although this "reservation" setting may lead to the under-utilization of storage resources in j^*, we believe this loss is acceptable in practice. Then, considering the case with the constraint (1c), we can simply take the expression of Ψ as $\max\{\lceil B_j / B_{j^*} \rceil\}$ and adopt the same method to work for it. As a conclusion, if the special central offices j^*_1, j^*_2 in two independent cases both reserve sufficient number of VMs (i.e., $\Psi_1 = \max\{\lceil D_j / D_{j^*} \rceil\}$, $\Psi_2 = \max\{\lceil B_j / B_{j^*} \rceil\}$), then we can declare that **the integral solution $\bar{\mathbf{z}}_t = \big\{ \max\{\bar{z}^1_{jt}, \bar{z}^2_{jt}\}, j \in \mathcal{N} \big\}$ does not violate the boxing constraint (1e) with probability 1**. Note that, this is a necessary condition of the feasible integral solution to the original problem. In addition, we present another necessary condition of the feasible integral solution to the original problem as shown in lemma 1.

Lemma 1: The integral solution $\bar{\mathbf{z}}_t$ derived by the RDRA satisfies

$$\begin{aligned} (i) \quad & \textstyle\sum_{j \in \mathcal{N}} D_j \bar{z}_{jt} \geq \sum_{c \in \mathcal{M}} \mathbb{1}_{\{n^c_{it} \geq 1, \exists i \in \mathcal{N}\}}, \\ (ii) \quad & \textstyle\sum_{j \in \mathcal{N}} B_j \bar{z}_{jt} \geq \sum_{i \in \mathcal{N}} \sum_{c \in \mathcal{M}} n^c_{it}, \end{aligned}$$

Proof: Since the ORFA can produce the near-optimal fractional solution $(\widetilde{\mathbf{x}}_t, \widetilde{\mathbf{y}}_t, \widetilde{\mathbf{z}}_t)$ for the problem $\mathbf{P}^t_2$, we can know that $\sum_j D_j \widetilde{z}_{jt} \geq \sum_j \sum_c \widetilde{y}^c_{jt} \geq \sum_j \sum_c \widetilde{x}^c_{ijt} \geq \sum_c \mathbb{1}_{\{n^c_{it} \geq 1, \exists i \in \mathcal{N}\}}$. In addition, we can also know that $\sum_j D_j \bar{z}_{jt} \geq \sum_j D_j \widetilde{z}_{jt}$ in terms of the weight conservation property of the RDRA. Combining them together, we can obtain the first part of the lemma. The second part can also be achieved with a similar procedure. ■

Finally, we declare that **the integral solution $\bar{\mathbf{z}}_t$ is a feasible solution to the original problem P_1 at each time frame**. Mathematically, the above necessary conditions cannot guarantee the feasibility of $\bar{\mathbf{z}}_t$. Nevertheless, as we assume the cooperative central offices are fully connected to provide the edge caching service[10] , we can prove it by contradiction.

Proof: For a clear description without duplication, we only take the case with the constraint (1d) into account. Suppose that there is no feasible solutions for $\mathbf{x}_t, \mathbf{y}_t$ when we bring the integral solution $\bar{\mathbf{z}}_t$ back to the original problem. In this context, we can know that at least one central office violates the constraint (1d) such as $D_{j_1} \bar{z}_{j_1 t} < \sum_c \bar{y}^c_{j_1 t}$, and the other central offices exactly satisfy that constraint such as $D_{j_2} \bar{z}_{j_2 t} = \sum_c \bar{y}^c_{j_2 t}$. The reasons are two-fold. First, since D_j indicates the maximum number of contents stored by one VM in a central office j, we believe it should be an

[10] The mobile network operator (or multiple cooperative ones) can offine designate qualified central offices to jointly provide the edge caching service. For example, it can model and quantize the relationship between each pair of central offices, then apply some community partition methods to generate a set of clusters, in each of which the central offices are well-connected.

integer value, otherwise the fractional part of VM storage resources is wasted in vain. Second, the fully connected central offices can make full use of their allocated storage resources as a whole. Therefore, it should not occur the case that $D_{j_1}\bar{z}_{j_1t} < \sum_c \bar{y}^c_{j_1t}$ while $D_{j_2}\bar{z}_{j_2t} > \sum_c \bar{y}^c_{j_2t}$, since j_2 can at least help to cache one content for j_1 until it is full. Summing up all the constraints, we can obtain $\sum_j D_j\bar{z}_{jt} < \sum_j \sum_c \bar{y}^c_{jt}$. However, since the ORFA can produce the feasible fractional solution $(\widetilde{\mathbf{x}}_t, \widetilde{\mathbf{y}}_t, \widetilde{\mathbf{z}}_t)$ for the problem $\mathbf{P}_2^t$, we can know that $\sum_j D_j\widetilde{z}_{jt} \ge \sum_j \sum_c \widetilde{y}^c_{jt}$. In addition, we can always obtain an integral solution $\bar{\mathbf{y}}'_t$ which satisfies $\sum_j \sum_c \bar{y}^{c'}_{jt} = \sum_j \sum_c \widetilde{y}^c_{jt}$ due to the totally unimodular property of the constraint matrix concerning $\boldsymbol{y}_t$, which will be further discussed in the next subsection (i.e., lemma 2 in Section IV-C). As $\bar{\mathbf{z}}_t$ is not a feasible solution for the original problem, we can know that $\bar{\mathbf{z}}_t$ cannot "fully cover" $\bar{\mathbf{y}}'_t$, otherwise $(\widetilde{\mathbf{x}}_t, \bar{\mathbf{y}}'_t, \bar{\mathbf{z}}_t)$ should be a feasible solution to the original problem. In other words, we have $\sum_j D_j\bar{z}_{jt} < \sum_j \sum_c \bar{y}^{c'}_{jt}$ in terms of the above discussions. According to these three inequations, we finally derive $\sum_j D_j\bar{z}_{jt} < \sum_j D_j\widetilde{z}_{jt}$, which contradicts the weight conservation property of the RDRA. ■

C. Dual Decomposition-based Optimal Algorithm

Motivation: We bring the feasible integral solution $\bar{\mathbf{z}}_t$ back to the original problem which consequently degrades to the following joint content placement and request routing problem $\mathbf{P}_3^t$ at each time frame:

$$
\begin{array}{lllr}
\min & \sum_i \sum_j \sum_c d_{ij} x^c_{ijt} n^c_{it} + \sum_j \sum_c b_j [y^c_{jt} - y^c_{jt-1}]^+ & & \\
\text{s.t.} & \sum_j x^c_{ijt} \ge \mathbb{1}_{\{n^c_{it} \ge 1\}}, & \forall i,\ \forall c, & \text{(2a)} \\
 & y^c_{jt} \ge x^c_{ijt}, & \forall i,\ \forall j,\ \forall c, & \text{(2b)} \\
 & B'_j \ge \sum_i \sum_c x^c_{ijt} n^c_{it}, & \forall j, & \text{(2c)} \\
 & D'_j \ge \sum_c y^c_{jt}, & \forall j. & \text{(2d)} \\
\text{var} & x^c_{ijt} \in [0,1],\ \ y^c_{jt} \in \{0,1\}. & & \text{(2e)}
\end{array}
$$

where $B'_j = B_j\bar{z}_{jt}$ and $D'_j = D_j\bar{z}_{jt}$. Note that, this problem is still a mixed-integer program which is difficult to solve directly. Inspired from the RDRA, a similar idea is to round the fractional solution $\widetilde{\mathbf{y}}_t$ generated by the ORFA to the integral solution $\bar{\mathbf{y}}_t$. Nevertheless, the RDRA cannot be applied to this problem directly and it is not easy to design a proper dependent rounding algorithm due to the packing constraints[11] (2c), (2d). Therefore, we need to pursue a new method rather than rounding $\widetilde{\mathbf{y}}_t$ directly. Since the control variables $\mathbf{x}_t$ and $\mathbf{y}_t$ are only coupled in the constraint (2b), we can invoke the dual-decomposition technique to design an optimal algorithm for this problem.

Algorithm Design: Thanks to the binary control variables $\mathbf{y}_t$, we first substitute $b_j[y^c_{jt} - y^c_{jt-1}]^+$ in the objective with a linear function $\phi^c_j(y,t)$ which takes the expression $b_j y^c_{jt}$ if $y^c_{jt-1} = 0$, and takes the value 0 if $y^c_{jt-1} = 1$. Then, we lift the constraint (2b) to the objective with a Lagrangian multiplier $\boldsymbol{\lambda}_t = \{\lambda^c_{ijt}, \forall i, j \in \mathcal{N}, \forall c \in \mathcal{M}\}$ as follows:

$$\sum_i \sum_j \sum_c (d_{ij} n^c_{it} + \lambda^c_{ijt}) x^c_{ijt} + \sum_j \sum_c (\phi^c_j(y,t) - \sum_i \lambda^c_{ijt} y^c_{jt})$$

Finally, we derive the Lagrangian dual problem of the problem $\mathbf{P}_3^t$, which can be separated into two independent subproblems:

$$
\begin{array}{lll}
\max & L(\boldsymbol{\lambda}_t) = L_1(\boldsymbol{\lambda}_t) + L_2(\boldsymbol{\lambda}_t) & \text{var } \lambda^c_{ijt} \ge 0, \\
\text{where} & L_1(\boldsymbol{\lambda}_t) = \min \sum_i \sum_j \sum_c (d_{ij} n^c_{it} + \lambda^c_{ijt}) x^c_{ijt} & \\
 & \qquad \text{s.t. (2a), (2c),} \quad \text{var } x^c_{ijt} \in [0,1], & \\
 & L_2(\boldsymbol{\lambda}_t) = \min \sum_j \sum_c (\phi^c_j(y,t) - \sum_i \lambda^c_{ijt} y^c_{jt}) & \\
 & \qquad \text{s.t. (2d),} \qquad \text{var } y^c_{jt} \in \{0,1\}. &
\end{array}
$$

To address this dual problem efficiently, we develop a dual decomposition-based optimal algorithm as shown in Alg. 3 via the subgradient method, which also produces the optimal results for the primal variable $\mathbf{x}_t, \mathbf{y}_t$ (i.e., the optimal solution for the problem $\mathbf{P}_3^t$).

Algorithm 3: DDOA

```
Input: $\boldsymbol{d}$, $\boldsymbol{n}_t$, $\boldsymbol{b}$, $\bar{\mathbf{z}}_t$, $\bar{\mathbf{y}}_{t-1}$, $\mathbf{D}$, $\mathbf{B}$, $\boldsymbol{\lambda}_t = \mathbf{0}$
Output: $\mathbf{x}^*_t, \bar{\mathbf{y}}_t$
Introduce $m$ to record the number of iterations, and initially set it to 1;
while termination criterions for the iteration (e.g., $m \ge 2000$) do not happen do
    Find optimal request routing decisions $\mathbf{x}^*_t$ to solve the subproblem $L_1(\boldsymbol{\lambda}_t)$;
    Find optimal content placement decisions $\bar{\mathbf{y}}_t$ to solve the subproblem $L_2(\boldsymbol{\lambda}_t)$;
    Update dual variables by
        $\lambda^c_{ijt} = \lambda^c_{ijt} + \Delta(m)(x^c_{ijt} - y^c_{jt}), \forall i, j \in \mathcal{N}, \forall c \in \mathcal{M}$;
    $m = m + 1$;
```

Specifically, given the dual variables $\boldsymbol{\lambda}_t$ in any an iteration m ($\boldsymbol{\lambda}_t = \mathbf{0}$ when $m = 1$ initially), the DDOA will independently solve the subproblem $L_1(\boldsymbol{\lambda}_t)$ and $L_2(\boldsymbol{\lambda}_t)$. As $L_1(\boldsymbol{\lambda}_t)$ is a standard linear program, we can invoke many mature methods (e.g., the simplex method) to solve it efficiently. Since $L_2(\boldsymbol{\lambda}_t)$ is an integer program, we do not solve it directly, but consider to address a linear program in which the integral constraint (5e) is relaxed (i.e., $y^c_{jt} \in [0,1]$). The following lemma shows that the optimal solution to the relaxed linear program is also the optimal solution to the original integer one.

Lemma 2: The optimal solution to the relaxed linear program of the original integer program is integral.

We leave the detail proofs of this lemma in the online technical report [34]. After solving two subproblems L_1, L_2 optimally, the DDOA will update the dual variables $\boldsymbol{\lambda}_t$ for the next iteration, with $\boldsymbol{\lambda}_t$ and the derived optimal solution $(\mathbf{x}^*_t, \bar{\mathbf{y}}_t)$ in the current iteration. Here, $x^c_{ijt} - y^c_{jt}$ indicates the subgradient direction and $\Delta(m)$ is a step size used in the m iteration. In this paper, we set it to U which initially is set to 1. Then, we set $U = U/2$, if the value of $L(\boldsymbol{\lambda}_t)$ does not increase after a number of iterations. In addition, we provide some termination criterions for the iteration (e.g., $m \ge 2000$ and $U \le 0.005$). Note that we choose the above settings due to their easy implementations in practice, while we can adopt other kinds of subgradient settings in our algorithm without affecting the performance greatly.

D. Summary

According to all the discussions in this section, we can conclude that the ORFA, RDRA and DDOA constitute the

[11] Intuitively, simply rounding all $\widetilde{y}_t$ up may violate the constraint (2d), while rounding them down (e.g., only one cache for each content) may violate the constraint (2c).

complete online algorithm which can always produce a feasible solution $(\mathbf{x}^*_t, \bar{\mathbf{y}}_t, \bar{\mathbf{z}}_t)$ for the original problem $\mathbf{P}_1$ at each time frame. As for the time complexity of the proposed algorithm, we first consider that the ORFA has a polynomial running time since we can exploit the primal-dual interior point method to produce the ϵ_{tol}-solution (almost optimal when ϵ_{tol} is set to a sufficiently small value such as $\epsilon_{tol} = 10^{-8}$) for the standard convex optimization problem $\mathbf{P}^t_2$ at each time frame [32], [33]. Second, the RDRA's complexity is $\mathcal{O}(N)$, since this algorithm is to round the fractional $\widetilde{\mathbf{z}}_t = \{\widetilde{z}_{jt}, \forall j \in \mathcal{N}\}$ to integral $\bar{\mathbf{z}}_t$. Third, the DDOA has a polynomial running time, since it has limited iterations due to the termination criterions and the two subproblems L_1, L_2 in each iteration are standard linear programs. To sum up, we can declare that our online algorithm achieves a polynomial running time.

V. Theoretical Performance Analysis

In this section, we will study the theoretical performance of our online algorithm in terms of the *competitive ratio*. Our key idea as depicted in Fig. 2 is to establish the following chain:

$$\begin{aligned}\text{OPT}(\mathbf{P}_1) &\geq \text{OPT}(\mathbf{P}'_1) \geq \mathbf{P}_4(\{\pi(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)|\forall t\}) \\ &\geq \frac{1}{r_1}\mathbf{P}_1(\{\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t|\forall t\}) \geq \frac{1}{r_1 r_2}\mathbb{E}\big[\mathbf{P}_1(\{\boldsymbol{x}^*_t, \bar{\boldsymbol{y}}_t, \bar{\boldsymbol{z}}_t|\forall t\})\big],\end{aligned}$$

where we denote by $\mathbf{P}_i(x)$ the objective value of problem i evaluated at point x, by OPT$(\cdot)$ the offline optimal objective value, and by $\mathbb{E}[\cdot]$ the expectation value. Note that, we consider the expected performance due to the *randomized* dependent rounding in our online algorithm.

To begin with, as $\mathbf{P}_1$ is a minimization problem and its integral constraints of control variables are relaxed in $\mathbf{P}'_1$, we can easily achieve OPT$(\mathbf{P}_1) \geq$ OPT$(\mathbf{P}'_1)$, since the latter has a larger solution space. In addition, we can obtain OPT$(\mathbf{P}'_1) \geq$ OPT$(\mathbf{P}_4)$ due to weak duality, and OPT$(\mathbf{P}_4) \geq \mathbf{P}_4(\{\pi(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)|\forall t\})$ since the dual problem $\mathbf{P}_4$ is a maximization problem, which consequently generates OPT$(\mathbf{P}'_1) \geq \mathbf{P}_4(\{\pi(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)|\forall t\})$. In this context, we next desire to walk through the following steps to achieve the whole chain:

S1: Construct a mapping π to produce a feasible solution to $\mathbf{P}_4$, in terms of the near-optimal fractional solution $(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)$ of $\mathbf{P}_2$ at each time frame;

S2: Prove $\mathbf{P}_1(\{\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t|\forall t\}) \leq r_1\mathbf{P}_4(\{\pi(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)|\forall t\})$. r_1 is considered as the interim competitive ratio;

S3: Prove $\mathbb{E}\big[\mathbf{P}_1(\{\boldsymbol{x}^*_t, \bar{\boldsymbol{y}}_t, \bar{\boldsymbol{z}}_t|\forall t\})\big] \leq r_2\mathbf{P}_1(\{\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t|\forall t\})$. r_2 is considered as the rounding gap, which combining with r_1 produces the final competitive ratio $r = r_1 r_2$;

S4: In the end, we will make some discussions about the derived competitive ratio r.

A. Feasible Mapping π

As for the **S1**, we first need to derive the dual problem $\mathbf{P}_4$ of the relaxed problem $\mathbf{P}'_1$, while this is not easy to achieve, due to the time-correlated adjustment costs and the boxing constraints of control variables. To this end, we introduce an equivalent formulation $\mathbf{F}$ for the relaxed problem:

$$\begin{aligned}
\min \quad & \mathbf{F} = \sum_t\sum_j c_{jt}z_{jt} + \sum_t\sum_i\sum_j\sum_c d_{ij}x^c_{ijt}n^c_{it} \\
& \qquad + \sum_t\sum_j s_j u_{jt} + \sum_t\sum_j\sum_c b_j v^c_{jt} \\
\text{s.t.} \quad & \text{(1a)} - \text{(1d)}, \\
& u_{jt} \geq z_{jt} - z_{jt-1}, \ \forall j, \forall t, && \text{(3a)} \\
& v^c_{jt} \geq y^c_{jt} - y^c_{jt-1}, \ \forall j, \forall c, \forall t, && \text{(3b)} \\
& \textstyle\sum_{j\in\mathcal{N}\setminus\mathcal{N}'} B_j z_{jt} \geq B_t - \sum_{j\in\mathcal{N}'} B_j C_j, \\
& \forall \mathcal{N}' \subset \mathcal{N} \cap B_t > \textstyle\sum_{j\in\mathcal{N}'} B_j C_j, && \text{(3c)} \\
& \textstyle\sum_{j\in\mathcal{N}\setminus\mathcal{N}'} D_j z_{jt} \geq D_t - \sum_{j\in\mathcal{N}'} D_j C_j, \\
& \forall \mathcal{N}' \subset \mathcal{N} \cap D_t > \textstyle\sum_{j\in\mathcal{N}'} D_j C_j, && \text{(3d)} \\
\text{var} \quad & x^c_{ijt} \geq 0, \ \ y^c_{jt} \geq 0, \ \ z_{jt} \geq 0, \ \ u_{jt} \geq 0, \ \ v^c_{jt} \geq 0.
\end{aligned}$$

Note that, the introduced variables $\boldsymbol{u}_t$, $\boldsymbol{v}_t$ associated with the constraint (3a), (3b) are equivalent to the time-correlated adjustment costs, which do not change the fractional solution $(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)$. As for the boxing constraints, we find that there is no incentive to give any variable in $\boldsymbol{x}_t$ a value larger than 1, due to the constraint (1a), (1c) and the monotonicity of $\mathbf{F}(\boldsymbol{x}_t)$. Therefore, we can remove the boxing constraint of $\boldsymbol{x}_t$ directly (i.e., $x^c_{ijt} \in [0,1] \Rightarrow x^c_{ijt} \geq 0$). After that, we can remove the boxing constraint of $\boldsymbol{y}_t$, since $\boldsymbol{y}_t$ can always be reduced to the same values with $\boldsymbol{x}_t$ (i.e., the precedence constraint (1b)), and similarly remove the boxing constraint of $\boldsymbol{v}_t$. Note that, we cannot tackle the boxing constraint of $\boldsymbol{z}_t$ in a similar way, due to the covering constraint (1c), (1d). Instead, we can replace it by a set of knapsack cover (KC) constraints (i.e., the constraint (3c), (3d)) as suggested by [35]. Here, $B_t \triangleq \sum_i\sum_c n^c_{it}$ and $D_t \triangleq \sum_c \mathbb{1}_{\{n^c_{it}\geq 1, \exists i\in\mathcal{N}\}}$. Obviously, each KC constraint is trivial if the right-hand side is negative. After that, we can remove the boxing constraint of $\boldsymbol{u}_t$ and will further derive the dual problem $\mathbf{P}_4$ from $\mathbf{F}$. To avoid the huge number of dual variables caused by KC constraints messing up the following analysis, we only consider the constraint (3d) with $|\mathcal{N}'| = 1$ in the dual problem formulation[12]:

$$\textstyle\sum_{j\in\mathcal{N}\setminus j'} D_j z_{jt} \geq D_t - D_{j'}C_{j'}, \ \forall j' \in \mathcal{N} \cap D_t > D_{j'}C_{j'}. \quad (4)$$

Dual Problem Formulation: We derive the dual problem $\mathbf{P}_4$ of the relaxed problem $\mathbf{P}'_1$ from $\mathbf{F}$, where α^c_{it}, β^c_{ijt}, γ_{jt}, θ_{jt}, ϕ_{jt}, τ^c_{jt}, ρ_{jt} are the corresponding dual variables for the constraint (1a) – (1d), (3a), (3b), (4):

$$\begin{aligned}
\min \quad & \mathbf{P}_4 = \sum_t\sum_i\sum_c \alpha^c_{it}\mathbb{1}_{\{n^c_{it}\geq 1\}} + \sum_t\sum_j (D_t - D_jC_j)\rho_{jt} \\
\text{s.t.} \quad & -\alpha^c_{it} + \beta^c_{ijt} + d_{ij}n^c_{it} + \theta_{jt}n^c_{it} \geq 0, \ \ \forall i, \forall j, \forall c, \forall t, && \text{(5a)} \\
& -\textstyle\sum_i \beta^c_{ijt} + \gamma_{jt} + \tau^c_{jt} - \tau^c_{jt+1} \geq 0, \ \ \forall j, \forall c, \forall t, && \text{(5b)} \\
& c_{jt} - D_j\gamma_{jt} - B_j\theta_{jt} - D_j(\textstyle\sum_l \rho_{lt} - \rho_{jt}) \\
& \qquad + \phi_{jt} - \phi_{jt+1} \geq 0, \qquad \forall j, \forall t, && \text{(5c)} \\
& s_j - \phi_{jt} \geq 0, \qquad \forall j, \forall t && \text{(5d)} \\
& b_j - \tau^c_{jt} \geq 0, \qquad \forall j, \forall c, \forall t, && \text{(5e)} \\
& \text{All the dual variables} \geq 0.
\end{aligned}$$

[12] Our analytic technique in the following indeed applies if one desires to work with all the KC constraints – new dual variables, KKT conditions, and corresponding terms in the bounding derivations just need to be added.

Characterizing Regularized Solution: As we regularize the problem $\mathbf{P}_1'$ (as well as the equivalent formulation $\mathbf{F}$) to the convex problem $\mathbf{P}_2$ with the objective transformation, we can know that the near-optimal ϵ_{tol}-solution $(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)$ of $\mathbf{P}_2^t$ generated by the primal-dual interior point method is within a sufficiently small neighborhood region of a KKT point. That is, the ϵ_{tol}-solution (almost) satisfies the following Karush-Kuhn-Tucker (KKT) conditions (page 567 of [32]):

$$
\begin{aligned}
&-\widetilde{\alpha}_{it}^c + \widetilde{\beta}_{ijt}^c + d_{ij}n_{it}^c + \widetilde{\theta}_{jt}n_{it}^c - \widetilde{\varsigma}_{ijt}^c = 0 && \text{(K1)}\\
&-\sum_i \widetilde{\beta}_{ijt}^c + \widetilde{\gamma}_{jt} + \tfrac{b_j}{\eta}\ln\tfrac{\widetilde{y}_{jt}^c+\varepsilon'}{\widetilde{y}_{jt-1}^c+\varepsilon'} - \widetilde{\varphi}_{jt}^c = 0 && \text{(K2)}\\
&c_{jt} - D_j\widetilde{\gamma}_{jt} - B_j\widetilde{\theta}_{jt} - D_j(\textstyle\sum_l \widetilde{\rho}_{lt} - \widetilde{\rho}_{jt})\\
&\qquad\qquad + \tfrac{s_j}{\sigma_j}\ln\tfrac{\widetilde{z}_{jt}+\varepsilon}{\widetilde{z}_{jt-1}+\varepsilon} - \widetilde{\nu}_{jt} = 0 && \text{(K3)}\\
\hline
&\widetilde{\alpha}_{it}^c(\textstyle\sum_j \widetilde{x}_{ijt}^c - \mathbb{1}_{\{n_{it}^c\ge 1\}}) = 0 && \text{(K4)}\\
&\widetilde{\beta}_{ijt}^c(\widetilde{y}_{jt}^c - \widetilde{x}_{ijt}^c) = 0 && \text{(K5)}\\
&\widetilde{\theta}_{jt}(B_j\widetilde{z}_{jt} - \textstyle\sum_c\sum_i \widetilde{x}_{ijt}^c n_{it}^c) = 0 && \text{(K6)}\\
&\widetilde{\gamma}_{jt}(D_j\widetilde{z}_{jt} - \textstyle\sum_c \widetilde{y}_{jt}^c) = 0 && \text{(K7)}\\
&\widetilde{\rho}_{jt}(\textstyle\sum_l D_l\widetilde{z}_{lt} - D_j\widetilde{z}_{jt} - D_t + D_jC_j) = 0 && \text{(K8)}\\
&\widetilde{\varsigma}_{ijt}^c\widetilde{x}_{ijt}^c = 0 && \text{(K9)}\\
&\widetilde{\varphi}_{jt}^c\widetilde{y}_{jt}^c = 0 && \text{(K10)}\\
&\widetilde{\nu}_{jt}\widetilde{z}_{jt} = 0 && \text{(K11)}\\
&\widetilde{\alpha}_{it}^c, \widetilde{\beta}_{ijt}^c, \widetilde{\theta}_{jt}, \widetilde{\gamma}_{jt}, \widetilde{\rho}_{jt}, \widetilde{\varsigma}_{ijt}^c, \widetilde{\varphi}_{jt}^c, \widetilde{\nu}_{jt} \ge 0 && \text{(K12)}
\end{aligned}
$$

TABLE I: KKT conditions of the fractional solution $(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)$

Here, $(\widetilde{\alpha}_{it}^c,\ \widetilde{\beta}_{ijt}^c,\ \widetilde{\gamma}_{jt},\ \widetilde{\theta}_{jt},\ \widetilde{\rho}_{jt})$ is the corresponding optimal solution to the dual problem of $\mathbf{P}_2^t$.

Mapping: We construct a mapping π which maps the primal and dual solution of $\mathbf{P}_2^t$ to a feasible solution of $\mathbf{P}_4$ at each time frame. We denote this constructed solution by $\pi(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t) \triangleq (\alpha_{it}^c, \beta_{ijt}^c, \gamma_{jt}, \theta_{jt}, \phi_{jt}, \tau_{jt}^c, \rho_{jt})$, in which $\alpha_{it}^c = \widetilde{\alpha}_{it}^c$, $\beta_{ijt}^c = \widetilde{\beta}_{ijt}^c$, $\gamma_{jt} = \widetilde{\gamma}_{jt}$, $\theta_{jt} = \widetilde{\theta}_{jt}$, $\phi_{jt} = \frac{s_j}{\sigma_j}\ln\frac{C_j+\varepsilon}{\widetilde{z}_{jt-1}+\varepsilon}$, $\tau_{jt}^c = \frac{b_j}{\eta}\ln\frac{1+\varepsilon'}{\widetilde{y}_{jt-1}^c+\varepsilon'}$, $\rho_{jt} = \widetilde{\rho}_{jt}$, and we can easily verify that it satisfies the constraint (5a) – (5e) while guarantees all the variables are non-negative. For example, we have (5b), since $-\sum_i \beta_{ijt}^c + \gamma_{jt} + \tau_{jt}^c - \tau_{jt+1}^c = -\sum_i \widetilde{\beta}_{ijt}^c + \widetilde{\gamma}_{jt} + \frac{b_j}{\eta}\ln\frac{\widetilde{y}_{jt}^c+\varepsilon'}{\widetilde{y}_{jt-1}^c+\varepsilon'} = \widetilde{\varphi}_{jt}^c \ge 0$ in term of (K2) and (K12). We have (5e), since $b_j \ge \frac{b_j}{\eta}\ln\frac{1+\varepsilon'}{\widetilde{y}_{jt-1}^c+\varepsilon'}$ in terms of $\eta = \ln(1+\frac{1}{\varepsilon'})$ as mentioned in Section IV-A. Therefore, $\pi(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)$ is a feasible solution of the dual problem $\mathbf{P}_4$ at each time frame. We should emphasize that, the near-optimal ϵ_{tol}-solution $(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)$ of $\mathbf{P}_2^t$ actually satisfies the perturbed KKT optimality conditions (page 567 of [32]). That is, the right-hand-side of (K4)–(K11) should be $\mathcal{O}(\epsilon_{tol})$ rather than 0. In this context, the above mapping still holds.

B. Interim Competitive Ratio r_1

We next conduct the **S2** to study the interim competitive ratio achieved by the derived fractional solution, in terms of the feasible mapping π and the KKT conditions (K1)–(K12). Specifically, we respectively bound the static costs and the dynamic costs in $\mathbf{P}_1(\{\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t|\forall t\})$, and sum them up as the interim competitive ratio.

Bounding the static costs: We omit the tilde symbol in the notations of the optimal primal and dual solution for clarity. To facilitate the following analysis, we first introduce two facts:

$$
\begin{aligned}
&p - q \le p\ln\tfrac{p}{q}, && \forall p,q>0, && \text{(6a)}\\
&(\textstyle\sum_n p_n)\ln\tfrac{\sum_n p_n}{\sum_n q_n} \le \textstyle\sum_n p_n\ln\tfrac{p_n}{q_n}, && \forall p,q>0. && \text{(6b)}
\end{aligned}
$$

In terms of these two facts, we have $\sum_t y_{jt}^c\ln\frac{y_{jt}^c+\varepsilon'}{y_{jt-1}^c+\varepsilon'} \ge 0$ and $\sum_t z_{jt}\ln\frac{z_{jt}+\varepsilon}{z_{jt-1}+\varepsilon} \ge 0$. We demonstrate the latter here, and the former can be shown analogously. Specifically, we rewrite its left-hand side $\sum_{t=1}^T z_{jt}\ln\frac{z_{jt}+\varepsilon}{z_{jt-1}+\varepsilon} = \sum_{t=1}^T (z_{jt}+\varepsilon)\ln\frac{z_{jt}+\varepsilon}{z_{jt-1}+\varepsilon} - \sum_{t=1}^T \varepsilon\ln\frac{z_{jt}+\varepsilon}{z_{jt-1}+\varepsilon}$, (Here T refers to any given time frame) and have the following inequations:

$$
\begin{aligned}
&\textstyle\sum_{t=1}^T (z_{jt}+\varepsilon)\ln\frac{z_{jt}+\varepsilon}{z_{jt-1}+\varepsilon} \ge \Big(\sum_{t=1}^T (z_{jt}+\varepsilon)\Big)\ln\frac{\sum_{t=1}^T (z_{jt}+\varepsilon)}{\sum_{t=1}^T (z_{jt-1}+\varepsilon)}\\
&\ge \textstyle\sum_{t=1}^T (z_{jt}+\varepsilon) - \sum_{t=1}^T (z_{jt-1}+\varepsilon) = z_{jT} - z_{j0} = z_{jT}, \quad \text{(7a)}\\
&-\textstyle\sum_{t=1}^T \varepsilon\ln\frac{z_{jt}+\varepsilon}{z_{jt-1}+\varepsilon} = -\sum_{t=1}^T \varepsilon\big[\ln(z_{jt}+\varepsilon) - \ln(z_{jt-1}+\varepsilon)\big]\\
&= \varepsilon\ln(z_{j0}+\varepsilon) - \varepsilon\ln(z_{jT}+\varepsilon) = \varepsilon\ln\frac{\varepsilon}{z_{jT}+\varepsilon} \ge -z_{jT}. \quad \text{(7b)}
\end{aligned}
$$

Note that $z_{j0} \equiv 0$ since our framework starts from the time frame $t=1$. The inequation (7a) follows from (6b) and (6a); (7b) follows from (6a). Then, we complete the proof by adding (7a) and (7b) up. In this context, we can bound the static costs:

$$
\begin{aligned}
&\textstyle\sum_t\sum_j c_{jt}z_{jt} + \sum_t\sum_i\sum_j\sum_c d_{ij}x_{ijt}^c n_{it}^c\\
&\le \textstyle\sum_t\sum_j \big[D_j\gamma_{jt} + B_j\theta_{jt} + D_j(\sum_l \rho_{lt} - \rho_{jt}) + \nu_{jt}\big]z_{jt}\\
&\quad + \textstyle\sum_t\sum_i\sum_j\sum_c (\alpha_{it}^c - \beta_{ijt}^c - \theta_{jt}n_{it}^c + \varsigma_{ijt}^c)x_{ijt}^c && \text{(8a)}\\
&= \textstyle\sum_t \Big[\sum_i\sum_j\sum_c \alpha_{it}^c x_{ijt}^c + \sum_j D_j(\sum_l \rho_{lt} - \rho_{jt})z_{jt}\\
&\quad + \textstyle\sum_j D_j\gamma_{jt}z_{jt} - \sum_i\sum_j\sum_c \beta_{ijt}^c x_{ijt}^c + \sum_j B_j\theta_{jt}z_{jt}\\
&\quad - \textstyle\sum_i\sum_j\sum_c \theta_{jt}^c n_{it}^c x_{ijt}^c\Big] && \text{(8b)}\\
&\le \textstyle\sum_t \Big[\sum_i\sum_j\sum_c \alpha_{it}^c x_{ijt}^c + \sum_j\sum_l D_j\rho_{lt}z_{jt} - \sum_j D_j\rho_{jt}z_{jt}\\
&\quad + \textstyle\sum_j\sum_c y_{jt}^c\gamma_{jt} - \sum_i\sum_j\sum_c \beta_{ijt}^c x_{ijt}^c\Big] && \text{(8c)}\\
&\le \textstyle\sum_t \Big[\sum_i\sum_c \alpha_{it}^c\mathbb{1}_{\{n_{it}^c\ge 1\}} + \sum_j \rho_{jt}\sum_l D_l z_{lt} - \sum_j D_j\rho_{jt}z_{jt}\\
&\quad + \textstyle\sum_j\sum_c y_{jt}^c\sum_i \beta_{ijt}^c - \sum_i\sum_j\sum_c \beta_{ijt}^c x_{ijt}^c\Big] && \text{(8d)}\\
&\le \textstyle\sum_t \Big[\sum_i\sum_c \alpha_{it}^c\mathbb{1}_{\{n_{it}^c\ge 1\}} + \sum_j \rho_{jt}(D_t - D_jC_j)\Big]\\
&= \mathbf{P}_4(\{\pi(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)|\forall t\}). && \text{(8e)}
\end{aligned}
$$

(8a) follows from (K1), (K3) and $\sum_t z_{jt}\ln\frac{z_{jt}+\varepsilon}{z_{jt-1}+\varepsilon} \ge 0$; (8b) follows from (K9) and (K11); (8c) follows from (K6), (K7) and $\sum_j D_j\rho_{jt}z_{jt} \ge 0$; (8d) follows from (K2), (K4) and $\sum_t y_{jt}^c\ln\frac{y_{jt}^c+\varepsilon'}{y_{jt-1}^c+\varepsilon'} \ge 0$; (8e) follows from (K5) and (K8), which is $\mathbf{P}_4(\{\pi(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)|\forall t\})$.

Bounding the dynamic costs: We define the following notations to facilitate our derivation: $\sigma_{max} \triangleq \max\{\sigma_j\}$, $C_{max} \triangleq \max\{C_j\}$ and $\delta_{jt}^* \triangleq \max\{D_j, \frac{B_j}{\sum_i\sum_c n_{it}^c}\}$. Then, we introduce $\mathcal{N}_{zt} \triangleq \{j|z_{jt} > z_{jt-1}\}$ and $\mathcal{N}_{yt}^c \triangleq \{j|y_{jt}^c > y_{jt-1}^c\}$.

We first bound the VM reconfiguration cost as follows:

$$\begin{aligned}
&\textstyle\sum_t \sum_j s_j[z_{jt} - z_{jt-1}]^+ = \sum_t \sum_{j\in\mathcal{N}_{zt}} s_j(z_{jt} - z_{jt-1}) \\
&\le \sigma_{max} \textstyle\sum_t \sum_{j\in\mathcal{N}_{zt}} \frac{s_j}{\sigma_j}(z_{jt}+\varepsilon)\ln\frac{z_{jt}+\varepsilon}{z_{jt-1}+\varepsilon} && \text{(9a)}\\
&\le \sigma_{max}(C_{max}+\varepsilon) \textstyle\sum_t \sum_{j\in\mathcal{N}_{zt}} \frac{s_j}{\sigma_j}\ln\frac{z_{jt}+\varepsilon}{z_{jt-1}+\varepsilon} && \text{(9b)}\\
&\le \sigma_{max}(C_{max}+\varepsilon) \textstyle\sum_t \sum_{j\in\mathcal{N}_{zt}} \big[-c_{jt}+D_j\gamma_{jt}+B_j\theta_{jt} \\
&\qquad\qquad +D_j(\textstyle\sum_l \rho_{lt}-\rho_{jt})+\nu_{jt}\big] && \text{(9c)}\\
&\le \sigma_{max}(C_{max}+\varepsilon) \textstyle\sum_t \sum_{j\in\mathcal{N}_{zt}} \delta^*_{jt}\big[\gamma_{jt} \\
&\qquad\qquad +\theta_{jt}\textstyle\sum_i \sum_c n^c_{it}+\sum_l \rho_{lt}\big]+\nu_{jt} && \text{(9d)}\\
&\le \sigma_{max}(C_{max}+\varepsilon) \textstyle\sum_t \sum_{j\in\mathcal{N}_{zt}} \big\{\delta^*_{jt}\big[\sum_c \gamma_{jt} \\
&\qquad\qquad +\theta_{jt}\textstyle\sum_i \sum_c n^c_{it}+\sum_j \rho_{jt}\big]+\nu_{jt}\big\} && \text{(9e)}\\
&\le \sigma_{max}(C_{max}+\varepsilon) \textstyle\sum_t \sum_{j\in\mathcal{N}_{zt}} \big\{\delta^*_{jt}\big[\sum_c \sum_i \beta_{ijt} \\
&\qquad\qquad +\textstyle\sum_i \sum_c \theta_{jt} n^c_{it}+\sum_j \rho_{jt}\big]+\nu_{jt}\big\} && \text{(9f)}\\
&\le \sigma_{max}(C_{max}+\varepsilon) \textstyle\sum_t \sum_{j\in\mathcal{N}_{zt}} \big\{\delta^*_{jt}\big[\sum_i \sum_c \alpha^c_{it}\mathbb{1}_{\{n^c_{it}\ge 1\}} \\
&\qquad\qquad +\textstyle\sum_j (D_t - D_jC_j)\rho_{jt}\big]+\nu_{jt}\big\} && \text{(9g)}\\
&\le r_{11}\mathbf{P}_4(\{\pi(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)|\forall t\}), && \text{(9h)}
\end{aligned}$$

where $r_{11} = \sigma_{max}(C_{max} + \varepsilon)\max_t \sum_{j\in\mathcal{N}_{zt}} \delta^*_{jt}$. (9a) follows from (6a) and the definition of σ_{max}; (9b) follows from the definition of C_{max}; (9c) follows from (K3); (9d) follows from the definition of δ^*_{jt} and the elimination of non-positive terms from (9c); (9e) follows from the multiplication of the non-negative term γ_{jt} and a simple suffix substitution (i.e., $l \to j$); (9f) follows from (K2) and $\sum_t y^c_{jt}\ln\frac{y^c_{jt}+\varepsilon'}{y^c_{jt-1}+\varepsilon'} \ge 0$; (9g) follows from (K1) and two inequations: $\alpha^c_{it}\mathbb{1}_{\{n^c_{it}\ge 1\}} = \alpha^c_{it}$ and $D_t - D_jC_j \ge 1$. The former one is straightforward, since both α^c_{it} and the primal constraint (1a) are trivial and can be removed if $n^c_{it} = 0$. The reasons to support the latter one are two-fold. First, both D_j and C_j should be integer values to avoid the waste of storage resources as mentioned in the proof above Section IV-C. Second, we can know that $D_t > D_jC_j$, due to the definition of "non-trivial" KC constraints as indicated in (4). Putting them together, we can make sure $D_t - D_jC_j \ge 1$; (9h) follows because $\nu_{jt} = 0$. The reason is that $z_{jt} > z_{jt-1} \ge 0$ for each $y \in \mathcal{N}_{zt}$ and meanwhile $\nu_{jt}z_{jt} = 0$ in terms of (K11).

We next bound the content migration cost as follows:

$$\begin{aligned}
&\textstyle\sum_t \sum_j \sum_c b_j[y^c_{jt}-y^c_{jt-1}]^+ = \sum_t \sum_{j\in\mathcal{N}_{yt}} \sum_c b_j(y^c_{jt}-y^c_{jt-1}) \\
&\le \eta(1+\varepsilon')\textstyle\sum_t \sum_{j\in\mathcal{N}_{zt}} \sum_c \frac{b_j}{\eta}\ln\frac{y^c_{jt}+\varepsilon'}{y^c_{jt-1}+\varepsilon'} && \text{(10a)}\\
&\le \eta(1+\varepsilon')\textstyle\sum_t \sum_{j\in\mathcal{N}_{zt}} \sum_c \big\{\sum_i \beta^c_{ijt}+\varphi^c_{jt}\big\} && \text{(10b)}\\
&\le \eta(1+\varepsilon')\textstyle\sum_t \sum_{j\in\mathcal{N}_{zt}} \sum_c \big\{\sum_i \alpha^c_{it}\mathbb{1}_{\{n^c_{it}\ge 1\}}+\varphi^c_{jt}\big\} && \text{(10c)}\\
&\le r_{12}\mathbf{P}_4(\{\pi(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)|\forall t\}), && \text{(10d)}
\end{aligned}$$

where $r_{12} = \eta(1 + \varepsilon')\max_t |\mathcal{N}_{zt}|$. (10a) – (10d) omit the details, as they follow highly analogous derivations as (9a) – (9h). In particular, the involved KKT conditions are (K1), (K2) and (K10). To sum up, we can obtain the interim competitive ratio $r_1 = 1 + r_{11} + r_{22}$.

We should emphasize that, if we take the perturbed KKT optimality conditions into account, we can easily obtain (8e)′ = $\big[1+7\mathcal{O}(\epsilon_{tol})\big]\mathbf{P}_4(\{\pi(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)|\forall t\})$ with the same analytical procedures as discussed above, since we use (K4) – (K9) and (K11) in the analysis and the value of our objective is much larger than 1. In addition, we consider that the control variables with small enough values (e.g., lower than 10^{-3}) can be negligible in practice. That is, $j \notin \mathcal{N}_{zt}$ if $z_{jt} \le \Omega$, where Ω is the threshold (e.g., 10^{-3}) that judges whether the values of control variables are small enough. In this context, we can know that $\nu_{jt} \le \frac{1}{\Omega}\mathcal{O}(\epsilon_{tol})$ for all $j \in \mathcal{N}_{zt}$, in terms of (K11). Then, we can obtain $r'_{11} = \sigma_{max}(C_{max}+\varepsilon)\max_t \sum_{j\in\mathcal{N}_{zt}} \big[\delta^*_{jt}+\frac{1}{\Omega}\mathcal{O}(\epsilon_{tol})\big]$ with the same analytical procedures as discussed above, since the value of our objective is much larger than 1. Similarly, we can obtain $r'_{12} = \eta(1 + \varepsilon')\max_t \big[|\mathcal{N}_{zt}| + \frac{1}{\Omega}\mathcal{O}(\epsilon_{tol})\big]$. Note that since ϵ_{tol} is generally set to a very small value (e.g., 10^{-8}) in the primal-dual interior point method [32], [33], we can approach the results derived by the exact KKT conditions sufficiently close (i.e., (8e) ≈ (8e)′, $r_{11} \approx r'_{11}$ and $r_{12} \approx r'_{12}$).

C. Rounding Gap r_2

We next conduct the **S3** to explore the rounding gap r_2 achieved between the fractional solution $(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)$ and the final integral solution $(\boldsymbol{x}^*_t, \bar{\boldsymbol{y}}_t, \bar{\boldsymbol{z}}_t)$. Our basic idea is to exploit the relationship between $\widetilde{\boldsymbol{z}}_t$ and $\bar{\boldsymbol{z}}_t$ (i.e., marginal distribution property and weight conservation property) to establish the connection between their storage costs $\sum_j c_{jt}\widetilde{z}_{jt}$ and $\mathbb{E}\big[\sum_j c_{jt}\bar{z}_{jt}\big]$, then to take this connection as a bridge to bound the other costs in the objective.

The connection between $\sum_j c_{jt}\widetilde{z}_{jt}$ and $\mathbb{E}\big[\sum_j c_{jt}\bar{z}_{jt}\big]$: Similar to the discussions in Section IV-B, we first take the case with the constraint (1d) into account, and denote the special central office by j^*. According to the marginal distribution property, we have

$$\begin{aligned}
&\mathbb{E}\big[\textstyle\sum_{j\in\mathcal{N}_{\mathbb{R}t}} c_{jt}\bar{z}_{jt}\big] = \textstyle\sum_{j\in\mathcal{N}_{\mathbb{R}t}} c_{jt}\big(\mathbb{E}[X_j]+\lfloor z_{jt}\rfloor\big)\\
&\quad = \textstyle\sum_{j\in\mathcal{N}_{\mathbb{R}t}} c_{jt}\big(p_j+\lfloor z_{jt}\rfloor\big) = \sum_{j\in\mathcal{N}_{\mathbb{R}t}} c_{jt}\widetilde{z}_{jt}.
\end{aligned}$$

Besides, we obviously hold $\sum_{j\in\mathcal{N}_{\mathbb{Z}t}} c_{jt}\widetilde{z}_{jt} = \sum_{j\in\mathcal{N}_{\mathbb{Z}t}} c_{jt}\bar{z}_{jt}$ due to the definition of $\mathcal{N}_{\mathbb{Z}t}$. Therefore, we can know that $\mathbb{E}\big[\sum_{j\in\mathcal{N}\backslash j^*} c_{jt}\bar{z}_{jt}\big] = \sum_{j\in\mathcal{N}\backslash j^*} c_{jt}\widetilde{z}_{jt} \le \sum_j c_{jt}\widetilde{z}_{jt}$.

According to the weight conservation property ($\omega_j = D_j$ in our considered case), we have

$$\begin{aligned}
&D_{j^*}\bar{z}_{j^*} = D_{j^*}\bar{p}_{j^*} + D_{j^*}\lfloor\widetilde{z}_{j^*}\rfloor\\
&\quad \le (1+\Psi)D_{j^*} + D_{j^*}p_{j^*} + \textstyle\sum_{j\in\mathcal{N}_{\mathbb{R}t}} D_jp_j + D_{j^*}\lfloor\widetilde{z}_{j^*}\rfloor\\
&\quad \le (1+\Psi)D_{j^*} + \textstyle\sum_j D_j\widetilde{z}_{jt} \le (1+\kappa_t)\sum_j D_j\widetilde{z}_{jt}. \qquad (11)
\end{aligned}$$

The first inequation follows from the weight conservation property, the second one follows since we add some non-negative terms at the right-hand-side, and the third one follows due to the introduced parameter $\kappa_t \triangleq \frac{(1+\Psi)D_{j^*}}{\sum_c \mathbb{1}_{\{n^c_{it}\ge 1,\, \exists\, i\in\mathcal{N}\}}}$. Note that, $(1 + \Psi)D_{j^*} \le \kappa_t\sum_j D_j\widetilde{z}_{jt}$ in terms of the proof in Lemma 1. In this context, we can easily obtain

$$\begin{aligned}
&c_{j^*}\bar{z}_{j^*} = \tfrac{c_{j^*}}{D_{j^*}}D_{j^*}\bar{z}_{j^*} \le \tfrac{c_{j^*}}{D_{j^*}}(1+\kappa_t)\textstyle\sum_j D_j\widetilde{z}_{jt}\\
&\qquad \le (1+\kappa_t)\textstyle\sum_j \frac{c_{jt}}{D_j}D_j\widetilde{z}_{jt} = (1+\kappa_t)\sum_j c_{jt}\widetilde{z}_{jt}.
\end{aligned}$$

The first inequation follows from (11) and the second inequation follows from the definition of the special central office (i.e., $j^* = \arg\min \frac{c_{jt}}{D_j}, j \in \mathcal{N}$). Based on the above discussions, we can obtain $\mathbb{E}\big[\sum_j c_{jt}\bar{z}_{jt}\big] = \mathbb{E}\big[\sum_{j\in\mathcal{N}\backslash j^*} c_{jt}\bar{z}_{jt}\big] + c_{j^*}\bar{z}_{j^*} \le (2+\kappa_t)\sum_j c_j\widetilde{z}_{jt}$. Considering the other case with the constraint (1c) together, we finally obtain $\mathbb{E}\big[\sum_j c_{jt}\bar{z}_{jt}\big] \le r_2'\sum_j c_j\widetilde{z}_{jt}$, where $\bar{\mathbf{z}}_t = \big\{\max\{\bar{z}^1_{jt}, \bar{z}^2_{jt}\}, j \in \mathcal{N}\big\}$ and $r_2' = (4+\kappa^1_t+\kappa^2_t)$.

Bounding the other costs: According to the constraint (1c), (1d) and the relationship $\mathbb{E}\big[\sum_j c_{jt}\bar{z}_{jt}\big] \le r_2'\sum_j c_j\widetilde{z}_{jt} \le r_2'\mathbf{P}_1(\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t)$, we can easily obtain the followings:

$$\begin{aligned}
&\textstyle\sum_t \mathbb{E}\big[\sum_j s_j[\bar{z}_{jt}-\bar{z}_{jt-1}]^+\big] \\
&\qquad \le \max_{j,t}\tfrac{s_j}{c_{jt}} r_2'\mathbf{P}_1(\{\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t|\forall t\}) \qquad (12a)\\
&\textstyle\sum_t \mathbb{E}\big[\sum_j\sum_c b_j[\bar{y}^c_{jt}-\bar{y}^c_{jt-1}]^+\big] \\
&\qquad \le \max_{j,t}\tfrac{b_jD_j}{c_{jt}} r_2'\mathbf{P}_1(\{\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t|\forall t\}) \qquad (12b)\\
&\textstyle\sum_t \mathbb{E}\big[\sum_i\sum_j\sum_c d_{ij}x^{c^*}_{ijt}n^c_{it}\big] \\
&\qquad \le \max_{i,j} d_{ij}\max_{j,t}\tfrac{B_j}{c_{jt}} r_2'\mathbf{P}_1(\{\widetilde{\boldsymbol{x}}_t, \widetilde{\boldsymbol{y}}_t, \widetilde{\boldsymbol{z}}_t|\forall t\}) \qquad (12c)
\end{aligned}$$

To sum up, we can obtain the rounding gap $r_2 = \big[1 + \max_{j,t}\frac{s_j+b_jD_j+B_j\max_{i,j}d_{ij}}{c_{jt}}\big]r_2'$.

D. Discussions

In the following, we will make some discussions about the final competitive ratio $r = r_1r_2$:

$$\begin{aligned}
r_1 &= 1+\ln(1+\tfrac{C_{max}}{\varepsilon})(C_{max}+\varepsilon)\max_t\textstyle\sum_{j\in\mathcal{N}_{zt}}\delta^*_{jt} \\
&\quad +\ln(1+\tfrac{1}{\varepsilon'})(1+\varepsilon')\max_t|\mathcal{N}_{zt}|,\\
r_2 &= \big[1+\max_{j,t}\tfrac{s_j+b_jD_j+B_j\max_{i,j}d_{ij}}{c_{jt}}\big](4+\kappa^1_t+\kappa^2_t),
\end{aligned}$$

where $\delta^*_{jt} \triangleq \max\{D_j, \frac{B_j}{\sum_i\sum_c n^c_{it}}\}$, $\kappa^1_t \triangleq \frac{(1+\Psi_1)D_{j^*}}{\sum_c \mathbb{1}_{\{n^c_{it}\ge 1,\, \exists\, i\in\mathcal{N}\}}}$ and $\kappa^2_t \triangleq \frac{(1+\Psi_2)B_{j^*}}{\sum_i\sum_c n^c_{it}}$.

First, our competitive ratio decreases with the increase of total user requests (i.e., $\sum_i\sum_c n^c_{it}$), in term of the definition of $\delta^*_{jt}, \kappa^1_t, \kappa^2_t$. That is, our algorithm can well support various kinds of multimedia contents and their corresponding large number of user requests with performance guarantee, which makes the edge caching service be promising to alleviate the backhaul traffic pressure and meanwhile facilitates the ever-increasing mobile multimedia services.

Second, it does not increase linearly with the increase of joined center offices in term of the definition of $\mathcal{N}_{zt}$. This observation indicates that our algorithm advocates the cooperation among more center offices without reducing the performance greatly, which is conductive to "persuade" more mobile network operators to cooperatively participate in the edge caching service.

Third, it deceases with the increasing of the unit storage costs of central offices, which hints that our algorithm can still have good performance in the "resource shortage" period (e.g., mobile network operators normally will increase the unit storage cost if the available storage resources of central offices at some time is restricted due to various service competitions).

At last, similar to those in many edge computing and caching works [10], [14], [18], our competitive ratio is proportional to the storage capacity of central offices (i.e., the product of the unit VM capacity D_j and the maximum available VM amount C_j in our framework), due to the inherent difficulty such as the unique "triangular covering" feature and the boxing constraints of control variables in our problem. This observation is easy to understand since the storage capacity will determinate the solution space of offline optimal decision. Note that we consider the product of D_j and C_j is not too large to produce a very loose competitive ratio in our framework. This is because that the value of C_j would be limited due to the coexistence of multiple edge services (e.g., the BBU pool, edge caching, edge computing, and etc.) in the inherently resource-restricted central offices (e.g., edge clouds or cloudlets). Also, we can consider a large content chunk as the unit content size, which consequently will decrease the value of D_j. Together with the discussions in Section IV-D, we believe that our online algorithm consisting of the OFRA, RDRA and DDOA is reasonably good with a provable competitive ratio and a polynomial running time. We hope it will provide some new insights for the edge caching research community, and will glad to see some improvements of our algorithm (e.g., a more tight competitive ratio) in the future.

VI. Numerical Evaluation

In this section, we will further evaluate the performance of our online algorithm with extensive trace-driven simulations. To the best of our knowledge, there is no open-access datasets involving user mobility, user requests for multimedia contents, cell sites, and central offices simultaneously. Therefore, we alternatively synthesize a simulation scenario to approximately capture their features via two real-world datasets.

A. Simulation Scenario

Central Offices: We exploit the Shanghai taxi trajectory dataset [36] to model users, cell sites and central offices. The main reasons are two-fold. First, taxi trajectory traces are widely used to model urban human mobility patterns and the regions with many taxi visiting records can be viewed as Points of Interest (PoIs) [37]. Second, most of PoIs (e.g., business area and office area) are covered by a series of cell towers from the cellular network planning perspective [38]. That is, PoIs can be broadly viewed as cell sites. Based on these principles, we use taxi GPS records to simulate users and define PoIs with the traffic heatmap to simulate cell sites. Besides, as each central office serves a group of cell sites in practice, we randomly select several PoIs as a group to simulate a central office for simplicity. Specifically, we select the data from date Feb 20, 2007 and depict the traffic heatmap as shown in Fig. 3, in terms of the every-minute GPS records of roughly 4000 taxis. In this figure, we take the grid (i.e., 0.6km×1km) whose heat level is over 300 as a cell site, and randomly assign 20 grids to a central office. We regard each taxi GPS record as a new user and set the length of time frame to 10 minutes. In this context, each central office can get the number of users in its service area at each time frame.

User Requests: Similarly to many edge caching works [11], [12], [28], we also use the YouTube video dataset [39] to generate user requests for contents, since it can well capture the time-varying feature of content popularity [3], [4] in terms

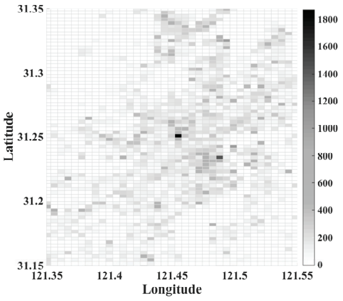


Fig. 3: Traffic heat.

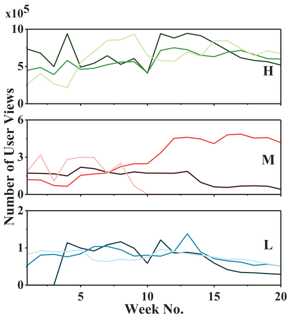


Fig. 4: Time-varying user views.

of user view times in different weeks. To proceed, we first rank the content popularity in terms of the accumulated number of user views, and respectively regard the top 50, 50 – 150, 150 – 200 as High, Middle and Low viewed contents. Then, we calculate their weekly user views (e.g., partial results are shown in Fig. 4), and scale them down for the afterward process (e.g., 100× smaller). We randomly distribute the scale-down user views of each content to the total "active users" which are within the grids of central offices at each time frame.

Capacity: We consider that the total resource capacity of all central offices should be slightly larger than the maximum requirement for the resource in the system. As for the content caching, we simply set the total content storage capacity as 2 times of the requested content amount (i.e., the input of simulation). As for the request routing, we record the maximum active user amount with different simulations as samples, and set the total user connection capacity as 150% of the average of the samples. Then, we distribute the total resource (i.e., storage and user connection) capacity to all the central offices proportionally in terms of their accumulated grid heat level. Next, we randomly pick a value from 5–10 for each central office as its maximum number of active VMs (i.e., C_j), and further calculates its VM instance setting (i.e., D_j, B_j), according to the quotient of its allocated resource capacity and maximum active VM amount.

Unit Costs: For the unit storage cost of a central office (e.g., c_{jt}), we leverage the same setting with our previous work [10], where the initial value is reversely proportional to the allocated storage capacity, and varies on the fly according to Gaussian distributions. For the unit request routing cost between central offices (e.g., d_{ij}), we consider the similar setting in [11], where the cost is indicated by the weighted delay between them. To this end, we assume a central office and its grid with the highest heat level share the same location, which is used to measure the delay by the geographical distance between central offices, and we introduce a specific coefficient for each pair of center offices to make sure the product of the delay and coefficient is roughly in the same order of magnitude with the unit storage cost. For two unit adjustment costs, we set them to be in the same order of magnitude with the unit storage cost. To achieve this goal, we can follow the expression of the competitive ratio r_2 to determinate the proper coefficients between the unit storage cost and the other unit costs.

B. Simulation Setup and Results

We implement the above simulation scenario with the open-source simulator ONE [40], describe our problem with AMPL and solve it with the IPOPT solver (i.e., the primal-dual interior point method) [33]. Since our online algorithm consists of multiple components, we will evaluate it with three groups of algorithms in terms of different perspectives.

In the first group, we will reveal the empirical competitive ratio r_1 and r_2, and therefore we take **F-OPT**, the offline optimal algorithm for the relaxed problem $\mathbf{P}_1'$; **One-shot**, the online algorithm that optimally solve the relaxed problem $\mathbf{P}_1'$ at each time frame; **ORFA**, the online approximation algorithm for the relaxed and regularized problem $\mathbf{P}_2$, which is the first component of ours; and **Ours**, the complete version of our online algorithm into account.

In the second group, we will study how the randomized dependent rounding algorithm would perform in our online algorithm, compared with **RDRA-Y**, the same algorithm as ours while it rounds $\widetilde{\boldsymbol{y}}_t$ rather than $\widetilde{\boldsymbol{z}}_t$. **IRA**, a randomized independent rounding algorithm where each control variable $\widetilde{z}_{jt}$ is rounded up or down independently of others; **CRA**, a conservative rounding algorithm where all the control variables $\widetilde{\boldsymbol{z}}_t$ are rounded up. Note that the other components of our online algorithm are still used in these three algorithms.

In the third group, we will evaluate the performance of our algorithm, compared with three alternative algorithms in the latest related works. **Matroid**, a matroid-based request routing algorithm which aims at caching content as much as possible to minimize the request routing cost [5], [28]; **Cache-only**, an approximate content placement algorithm which aims at minimizing the content caching and retrieving costs for the cooperative base stations [6]. We modify it to work for our problem, since it assumes BSs have unlimited storage capacity; **Greedy**, a greedy joint content placement and request routing algorithm, which aims at minimizing the content caching, request routing and content mitigation cost at each time frame [11], [12]. It is close to ours while it neither considers the resource allocation nor has the performance guarantee.

We adopt the algorithm competitive ratio as the performance metric, which measures the ratio between its performance and the offline algorithm's performance (here, the offline algorithm is for the original problem $\mathbf{P}_1$). We execute the following simulation, in which we randomly choose 20 continuous time frames (i.e., a 20-week record in the YouTube dataset) from 8am to 8pm in taxi trajectory traces as six independent test cases. In a test case, we genreate 10 central offices and randomly select 30 contents from one category (e.g., High viewed) for simulation. Each test case is repeated for ten times.

The average simulation results of three groups are shown in Fig. 5 – Fig. 7. Globally, we observe that our algorithm has a better performance in the case of the high user requests, which is in accordance with our theoretical analysis in Section V-D. In addition, it can achieve a low empirical competitive ratio r_1 and r_2 as shown in Fig. 5. For example, the performance of ORFA (i.e., r_1) can achieve 1.5× of its corresponding offline optimum (i.e., F-OPT) on average, and improve up to 45% compared with the One-shot optimal algorithm. Meanwhile,

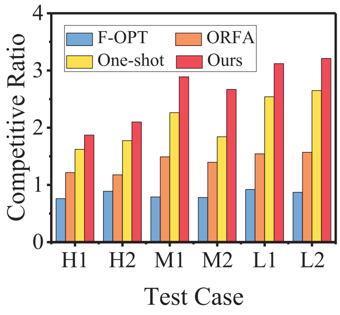


Fig. 5: Results of group 1.

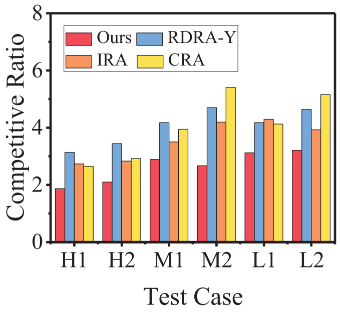


Fig. 6: Results of group 2.

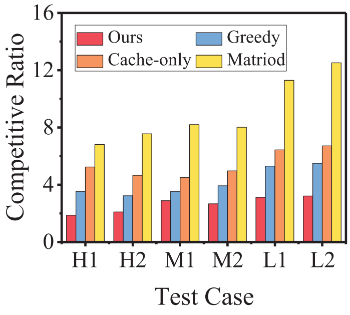


Fig. 7: Results of group 3.

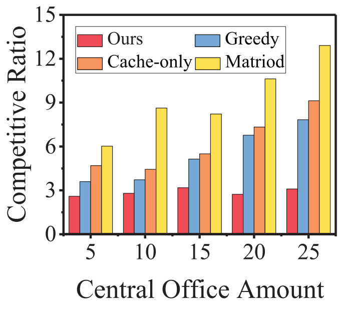


Fig. 8: Change of central office amount.

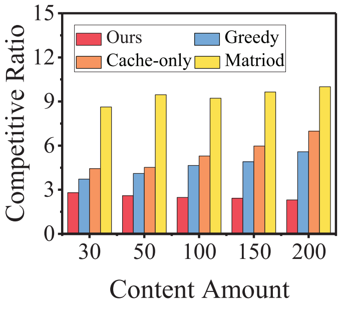


Fig. 9: Change of content amount.

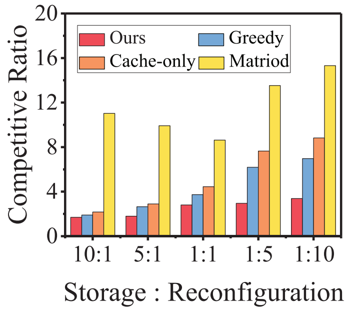


Fig. 10: Change of cost ratio.

the overall performance of our online algorithm (i.e., r_2) will be no more than 2.2× of ORFA's. These results validate the efficiency of each component in our online algorithm. Indeed, we can see in Fig. 6 that our dependent rounding algorithm has the best performance, and more importantly it can produce the feasible solution to the original problem with probability 1 while only 40% and 60% in IRA and RDRA-Y, respectively. In Fig. 7, the performance of Matroid algorithm is the worst as expected, since it tries to cache as much content as possible without taking the storage cost and the adjustment costs into account. The performance gap between Cache-only and Greedy emphasizes the effectiveness of request routing, especially when the number of user requests is large. Greedy performs better than Matroid and Cache-only, since it takes more system costs into account. However, we still notice a considerable gap to the offline optimum, which is mainly due to the overemphasis on one-shot optimization and no preparation for the future. In contrast, our regularization-based online algorithm can produce better results, which improves 40% performance compared with Greedy in most of cases.

In addition, we evaluate the flexibility of our online algorithm in terms of various system settings. First, we keep the same settings but change the number of central offices in test case M1 for simulation. The results in Fig. 8 show that increasing or decreasing the number of central offices has a slight impact on our performance, since our algorithm can make full use of the overall central office resources to serve user requests. As such, ours can improve up to 4 times, compared with Cache-only and Matroid. We next conduct a simulation to explore the relationship between algorithm performance and content amount, in which we generate 20 central offices and randomly select some contents from the top 300 popular contents in the dataset. Still, our performance is much better than the others. The reason is that the increase of content amount also brings in more resources in the system since the total resource capacity is always larger than the resource requirement as mentioned in the simulation setting, and the algorithm which can make full use of this additional resources will have a good performance. Also, we evaluate the performance with different ratios of unit storage cost and configuration cost. Since our regularization-based online algorithm emphasizes them equally in the objective, our performance is relatively stable with the increase of configuration cost. To sum up, we can declare that our online algorithm is flexible to support various system settings with "qualified" performance guarantee.

Central Office	GLPK	Total	ORFA	RDRA	DDOA
10	0.19s	0.24s	0.16	0.001	0.08
20	3.08s	0.93s	0.51	0.003	0.41
30	64.33s	1.83s	1.02	0.003	0.81
40	102.54s	6.26s	3.86	0.005	2.40
50	147.91s	11.48s	7.55	0.007	3.93

TABLE II: Running time with different central office amounts.

Content	GLPK	Total	ORFA	RDRA	DDOA
200	0.94s	0.36s	0.23	0.001	0.13
400	18.77s	1.11s	0.68	0.002	0.42
600	62.30s	2.25s	1.39	0.002	0.85
800	79.75s	3.68s	2.45	0.003	1.23
1000	115.88s	6.14s	4.02	0.004	2.12

TABLE III: Running time with different content amounts.

Next, we evaluate the practical running time of our algorithm regarding different numbers of central offices and contents. We exploit the anonym "GLPK" to indicate the running time of solving the mixed integer problem $\mathbf{P}_1$ by using the GLPK solver at each time frame, and "Total" to indicate the overall running time of our algorithm (i.e., the sum of ORFA, RDRA and DDOA). The running time of our algorithm with different central office amounts when the content amount is set to 30 is given in Table III, and that with different content amounts when the central office amount is set to 5 is given in Table IV. Note that these results are the average values in 50 evaluations with different input parameters, and the purpose of choosing the above small value 30 and 5 in the evaluations is to respectively highlight the "importance" of the number of central offices and contents in the running time.

We can find that, our algorithm is lightweight, for example it only consumes 6.14s when the content amount is 1000, which achieves $18.8\times$ speedup compared with the GLPK. As for each component of our algorithm, ORFA intuitively consumes more time since it has more constraints and control variables, and RDRA is negligible due to its low time complexity. In addition, we find that the time ratio of our algorithm is $11.48/0.24 = 47.8$ when the number of central offices increases 5 times, and that is 17.1 when the content amount increases 5 times. In other words, the number of central offices dominates the running time, which is in accordance with the intuition due to the control variables $x^c_{ijt},\ \forall i,j \in \mathcal{N}$. As a conclusion, since the number of central offices in practice is limited in the C-RAN based edge caching framework and our algorithm can well support a large number of contents, we believe that our proposed framework is efficiency, flexibility and lightweight.

At last, we also evaluate the impact of ϵ_{tol} (i.e, the parameter "compl_inf_tol" in the IPOPT solver [33]) on the algorithm performance and running time with the same settings as mentioned in Table II and III. For clarity, we omit the tedious results here but highlight our findings as follows: on the one hand, when this parameter is set to larger than 10^3, the algorithm performance will decrease greatly. For example, the total system cost when $\epsilon_{tol} = 10^4$ is on average 3.6x larger than that when $\epsilon_{tol} = 10^{-4}$; on the other hand, when it set to smaller than 10^{-10}, the algorithm running time will at least 2x longer than that when $\epsilon_{tol} = 10^{-4}$ without significant performance improvement. To summary, we will set ϵ_{tol} to $1 \sim 10^{-8}$ for our problem in practice, when jointly taking algorithm performance and running time into account.

VII. Conclusion

In this paper, we advocated the edge caching in C-RAN to facilitate the ever-increasing mobile multimedia services. We built a comprehensive model to capture the key components of edge caching in C-RAN, and formulated a joint optimization problem, aiming at minimizing the system costs in terms of storage, VM reconfiguration, content access latency, and content migration over time, and meanwhile satisfying the time-varying user requests and respecting various practical constraints. Then, we proposed a novel online approximation algorithm by resorting to the regularization, rounding and decomposition technique, which could be proved to achieve a parameterized competitive ratio and a polynomial running time. Extensive trace-driven simulations corroborated the efficiency, flexibility and lightweight of our online algorithm.

References

[1] "Cisco visual networking index: Global mobile data traffic forecast update, 2016 to 2021." Available in: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html.

[2] A. Checko, H. L. Christiansen, Y. Yan, *et al.*, "Cloud ran for mobile networks: A technology overview," *IEEE Communications surveys & tutorials*, vol. 17, no. 1, pp. 405–426, 2015.

[3] D. Liu, B. Chen, C. Yang, and A. F. Molisch, "Caching at the wireless edge: Design aspects, challenges, and future directions," *IEEE Communications Magazine*, vol. 54, no. 9, pp. 22–28, 2016.

[4] G. Paschos, E. Bastug, I. Land, *et al.*, "Wireless caching: Technical misconceptions and business barriers," *IEEE Communications Magazine*, vol. 54, no. 8, pp. 16–22, 2016.

[5] M. Dehghan, A. Seetharam, B. Jiang, *et al.*, "On the complexity of optimal routing and content caching in heterogeneous networks," in *IEEE Conference on Computer Communications*, pp. 936–944, 2015.

[6] A. Gharaibeh, A. Khreishah, B. Ji, and M. Ayyash, "A provably efficient online collaborative caching algorithm for multicell-coordinated systems," *IEEE Transactions on Mobile Computing*, vol. 15, no. 8, pp. 1863–1876, 2016.

[7] W. Jiang, G. Feng, and S. Qin, "Optimal cooperative content caching and delivery policy for heterogeneous cellular networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 5, pp. 1382–1393, 2017.

[8] X. Li, X. Wang, K. Li, *et al.*, "Collaborative multi-tier caching in heterogeneous networks: Modeling, analysis, and design," *IEEE Transactions on Wireless Communications*, vol. 16, no. 10, pp. 6926–6939, 2017.

[9] M. Lin, A. Wierman, L. L. Andrew, and E. Thereska, "Dynamic right-sizing for power-proportional data centers," *IEEE/ACM Transactions on Networking*, vol. 21, no. 5, pp. 1378–1391, 2013.

[10] L. Jiao, A. M. Tulino, J. Llorca, *et al.*, "Smoothed online resource allocation in multi-tier distributed cloud networks," *IEEE/ACM Transactions on Networking*, vol. 25, no. 4, pp. 2556–2570, 2017.

[11] Y. Wu, C. Wu, B. Li, *et al.*, "Scaling social media applications into geo-distributed clouds," *IEEE/ACM Transactions on Networking*, vol. 23, no. 3, pp. 689–702, 2015.

[12] L. Yang, J. Cao, G. Liang, and X. Han, "Cost aware service placement and load dispatching in mobile cloud systems," *IEEE Transactions on Computers*, vol. 65, no. 5, pp. 1440–1452, 2016.

[13] G. Ma, Z. Wang, M. Zhang, *et al.*, "Understanding performance of edge content caching for mobile video streaming," *IEEE Journal on Selected Areas in Communications*, vol. 35, no. 5, pp. 1076–1089, 2017.

[14] K. Poularakis, G. Iosifidis, A. Argyriou, *et al.*, "Caching and operator cooperation policies for layered video content delivery," in *IEEE Conference on Computer Communications*, pp. 1–9, 2016.

[15] K. Poularakis and L. Tassiulas, "Code, cache and deliver on the move: A novel caching paradigm in hyper-dense small-cell networks," *IEEE Transactions on Mobile Computing*, vol. 16, no. 3, pp. 675–687, 2017.

[16] S. Shukla and A. A. Abouzeid, "Proactive retention aware caching," in *IEEE Conference on Computer Communications*, pp. 1–9, 2017.

[17] S. Shukla, O. Bhardwaj, A. A. Abouzeid, *et al.*, "Hold'em caching: Proactive retention-aware caching with multi-path routing for wireless edge networks," in *ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 1–10, 2017.

[18] I. Hou, T. Zhao, S. Wang, *et al.*, "Asymptotically optimal algorithm for online reconfiguration of edge-clouds," in *International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 291–300, 2016.

[19] X. Qiu, H. Li, C. Wu, *et al.*, "Cost-minimizing dynamic migration of content distribution services into hybrid clouds," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 12, pp. 3330–3345, 2015.

[20] X. Zhang, C. Wu, Z. Li, and F. C. Lau, "Online cost minimization for operating geo-distributed cloud cdns," in *IEEE International Symposium on Quality of Service*, pp. 21–30, 2015.

[21] L. Jiao, L. Pu, L. Wang, *et al.*, "Multiple granularity online control of cloudlet networks for edge computing," in *IEEE International Conference on Sensing, Communication and Networking*, pp. 1–9, 2018.

[22] N. Yu, Y. Miao, L. Mu, *et al.*, "Minimizing energy cost by dynamic switching on/off base stations in cellular networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 11, pp. 7457–7469, 2016.

[23] S. Krishnasamy, P. Akhil, A. Arapostathis, *et al.*, "Augmenting max-weight with explicit learning for wireless scheduling with switching costs," in *IEEE Conference on Computer Communications*, 2017.

[24] L. Lu, J. Tu, C.-K. Chau, *et al.*, "Online energy generation scheduling for microgrids with intermittent energy sources and co-generation," in *ACM International Conference on Measurement and Modeling of Computer Science*, pp. 53–66, 2013.

[25] M. H. Hajiesmaili, C.-K. Chau, M. Chen, and L. Huang, “Online microgrid energy generation scheduling revisited: The benefits of randomization and interval prediction,” in *ACM International Conference on Future Energy Systems*, pp. 1–11, 2016.

[26] N. Buchbinder, S. Chen, and J. S. Naor, “Competitive analysis via regularization,” in *ACM-SIAM Symposium on Discrete Algorithms*, pp. 436–444, 2014.

[27] M. Chen, W. Saad, and M. Debbah, “Echo state networks for proactive caching in cloud-based radio access networks with mobile users,” *IEEE Transactions on Wireless Communications*, vol. 16, no. 6, pp. 3520–3535, 2017.

[28] T. X. Tran, A. Hajisami, and D. Pompili, “Cooperative hierarchical caching in 5g cloud radio access networks,” *IEEE Network*, vol. 31, no. 4, pp. 35–41, 2017.

[29] A. A. Ageev and M. I. Sviridenko, “Pipage rounding: A new method of constructing algorithms with proven performance guarantee,” *Journal of Combinatorial Optimization*, vol. 8, no. 3, pp. 307–328, 2004.

[30] R. Gandhi, S. Khuller, S. Parthasarathy, and A. Srinivasan, “Dependent rounding and its applications to approximation algorithms,” *Journal of the ACM*, vol. 53, no. 3, pp. 324–360, 2006.

[31] M. Dudík, S. J. Phillips, and R. E. Schapire, “Maximum entropy density estimation with generalized regularization and an application to species distribution modeling,” *Journal of Machine Learning Research*, vol. 8, no. 6, pp. 1217–1260, 2007.

[32] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[33] A. Wächter and L. T. Biegler, “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming,” *Springer Mathematical programming*, vol. 106, no. 1, pp. 25–57, 2006.

[34] “Online technical report.” Available in: https://www.dropbox.com/s/1caeqx6g16s8moh/JSAC2017_Online.pdf?dl=0.

[35] R. D. Carr, L. K. Fleischer, V. J. Leung, *et al.*, “Strengthening integrality gaps for capacitated network design and covering problems,” in *ACM-SIAM Symposium on Discrete Algorithms*, pp. 106–115, 2000.

[36] “Shanghai taxi trajectory traces.” Available in: http://wirelesslab.sjtu.edu.cn/taxi_trace_data.html.

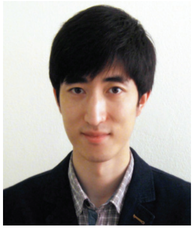
[37] X. Li, G. Pan, Z. Wu, *et al.*, “Prediction of urban human mobility using large-scale taxi traces and its applications,” *Springer Frontiers of Computer Science*, vol. 6, no. 1, pp. 111–121, 2012.

[38] F. Xu, Y. Li, H. Wang, *et al.*, “Understanding mobile traffic patterns of large scale cellular towers in urban environment,” *IEEE/ACM Transactions on Networking*, vol. 25, no. 2, pp. 1147–1161, 2017.

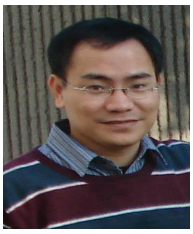
[39] “Statistics and social network of youtube videos.” Available in: http://netsg.cs.sfu.ca/youtubedata/.

[40] “The opportunistic network environment simulator.” Available in: https://akeranen.github.io/the-one/.

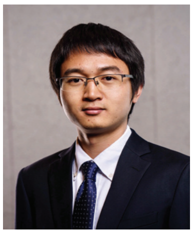
Lingjun Pu received the Ph.D. degree from Nankai University, China, in 2016. He was a joint Ph.D. student with University of Göttingen, Germany, from 2013 to 2015. He is currently an Assistant Professor with the College of Computer and Control Engineering, Nankai University. His research focuses on mobile cloud/edge computing, edge caching, 5G C-RAN, SDN/NFV and opportunistic routing.



Lei Jiao received the Ph.D. degree in computer science from University of Göttingen, Germany in 2014. He was a researcher at IBM Research in Beijing, China in 2010 prior to his Ph.D. study, and a member of technical staff at Bell Labs in Dublin, Ireland from 2014 to 2016. He is currently an Assistant Professor with the Department of Computer and Information Science, University of Oregon, USA. His research interests are broadly in the models, algorithms, and analysis for the optimization and control of distributed systems and networks.



Xu Chen (M’12) received the Ph.D. degree in information engineering from The Chinese University of Hong Kong in 2012. He was a Post-Doctoral Research Associate with Arizona State University, Tempe, USA, from 2012 to 2014, and a Humboldt Fellow with University of Göttingen, Germany, from 2014 to 2016. He is currently a Professor with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China. He received the 2014 IEEE INFOCOM Best Paper Runner-Up Award and the 2014 Hong Kong Young Scientist Award.



Lin Wang received the Ph.D. degree in computer science with distinction from the Institute of Computing Technology, Chinese Academy of Sciences in 2015. He was a visiting researcher at IMDEA Networks Institute, Madrid, Spain from 2012 to 2014, and a Research Associate at SnT Luxembourg from 2015 to 2016. Starting in July 2016, he has been head of the Smart Urban Networks research group in the Telecooperation Lab and has been an Athene Young Investigator since 2018, at TU Darmstadt, Germany. His current research interests include edge computing, networked systems, and energy-efficient algorithms.

Qinyi Xie is currently a Master student with the College of Computer and Control Engineering, Nankai University, China. Her research interests include Low-Power Wide-Area Networks (LPWAN), mobile edge computing and 5G C-RAN.

Jingdong Xu is currently a Professor with the College of Computer and Control Engineering, Nankai University, China. She is also the Head of the Computer Networks and Information Security Group. Her research interests include sensor networks, vehicle ad-hoc networks, network security and management, and opportunistic network and computing.