IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 8, 2021, accepted September 27, 2021, date of publication September 29, 2021,
date of current version October 7, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3116602

Dynamic Hierarchical Caching Resource Allocation for 5G-ICN Slice

LIPING GE, JINHE ZHOU, AND ZHENKANG ZHENG
Key Laboratory of the Ministry of Education for Optoelectronic Measurement Technology and Instrument, Beijing Information Science and Technology University, Beijing 100101, China
School of Information and Communication Engineering, Beijing Information Science and Technology University, Beijing 100101, China

Corresponding author: Jinhe Zhou (zhoujinhe@bistu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61872044 and Grant 61901043.

ABSTRACT Network slicing and Multiple-Access Edge Computing (MEC) are key technologies in fifth-generation (5G) networks. The flexible programmability of network slicing and the decentralization of MEC facilitate the deployment of Information-Centric Networking (ICN). The caching feature of ICN can provide users with low-latency data services. Although many existing works have addressed the cache deployment problem or the cache optimization problem, most of them do not consider the issue of caching resource allocation in the dynamic and hierarchical environment. Dynamic deployment of cache nodes can improve the operator's revenue as much as possible while accurately allocating the caching resources can reduce the user-requested latency. Therefore, in this study, a problem of the operator's expected revenue maximization is presented in an environment combining dynamic deployment of the MECs and the caching-enabled node ICN-Gateway (ICN-GW). To solve this problem, we propose an optimal stopping theory (OST)-based dynamic hierarchical caching resources allocation (ODH-CRA) algorithm. The algorithm consists of three parts. Firstly, an Integer Linear Programming (ILP) solution is proposed to determine the optimal deployment of the MECs. This method determines the optimal location and number of the MECs by considering deployment costs and service requirement costs synthetically. Secondly, a redeployment technique based on the OST is proposed to determine the best redeployment time of the MECs according to the values of latency violations and the service latency requirements. Finally, an improved elite genetic algorithm (IEGA) is proposed to find the optimal solution of the hierarchical caching resource allocation. This method searches the optimal scheme by maximizing the operator's revenue joint caching costs and energy consumption. Ultimately, we perform a series of simulation experiments to compare the proposed method's performance to dynamic and hierarchical methods. Our solution can effectively reduce the latency for users' requesting, improve the revenue of ICN Communication Service Provider (ICSP), and provide an effective caching resource allocation scheme for the next generation of Internet of Things (IoT) networks.

INDEX TERMS Network slice, MEC, ICN, hierarchical caching resource allocation, dynamic deployment.

I. INTRODUCTION

A very low latency communication environment plays a vital role in the Internet of Things (IoT). Network caching appears essential to accommodate the Quality of Experience (QoE) requirements of latency-sensitive applications. However, there is an evident mismatch between the expectation and implementation that existing IoT devices are intrinsically resource-constrained devices and cannot offer real-time scalable applications with minimal latency and high QoE. Now many researchers are committed to the research of the new generation IoT with Information-Centric Networking (ICN), Network slicing, and Multiple-Access Edge Computing (MEC) enabled in 5G [1]–[3].

5G, as an underlying technology, has an indispensable role to play for advancements of numerous technologies and services, IoT being one of them [4]. Significant differentiator operators seek in 5G is the transition toward a service-centric infrastructure. At present, many researchers are committed to enabling ICN service in 5G networks [5]–[7] since ICN is a content-centric service decoupling in time and space between publisher and subscriber [8]. ICN can introduce the named content as a network primitive and provide the in-network cache function. The design of caching schemes

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

has been attracting a lot of attention, and many researchers use game theory to solve the caching-related problem. M. Hajimirsadeghi *et al.* [9] developed an analytical framework in ICN and used game theory to study the caching strategy. The authors perform cache operation under the assumption that the caching cost is inversely proportional to popularity. The demand of some users can not be satisfied since caching cost is very high for these content with low popularity but higher quality. Y. Song *et al.* [10] applied cache function on the radio access network side. They paid attention to the joint optimization of latency and energy consumption of popular content by using the bargaining game method, which ensured good fairness. However, the author did not consider the importance of caching costs. To stimulate CPs, F. Shen *et al.* [11] proposed an incentive proactive caching mechanism based on game theory. The mechanism pays attention to caching costs while ignoring the influence of transmission energy consumption.

The optimal caching scheme has been studied, and the problem of offering the low-latency data service and the high-speed migration of content between the edge and the central cloud system is not yet resolved, for which MEC has been proposed. The typical characteristics of MEC technology include the closest proximity, ultra-low latency, multi-access function, and network context information [12]. This brings in the importance of the caliber of MEC. A. Ndikumana *et al.* [13] proposed a collaborative caching resource allocation and computing resource sharing scheme for the MECs. This method can reduce the number of data exchanges between users and the remote cloud, reducing network latency. However, there are many high-mobility 5G applications such as tactile Internet and autonomous vehicles. To further adapt to the needs of users with high mobility, R. Xie *et al.* [14] considered a hierarchical caching architecture that core network and radio access network (RAN) have the caching capability in 5G networks and studied the problem of hierarchical caching resource sharing for the mobile virtual network operators (MVNOs). Y. K. Tun *et al.* [15] examined a two-level resource allocation problem while used the Kelly mechanism to enable efficient resource utilization and maximized the total valuation of MVNOs. However, they ignored the latency requirement of the ICN-UE. Z. Zhang *et al.* [16] proposed a novel hierarchical proactive caching approach to relieve the high latency caused by users' high speed. The approach could determine the location of the video segment and proactively cache videos by predicting users' future demands. However, they ignored the influence of operators' revenue.

The expected revenue of operators cannot be ignored. This parameter ensures that users can obtain reliable data and improve this parameter by reasonably deploying storage resources. In the 5G network, network slicing can divide a physical network into multiple virtual networks consist of a series of virtual network functions (VNFs) to support ICN service requirements [17]. H. Jin *et al.* [18] proposed a heuristic method on the virtual cache function deployment from the point of view of ICN and service provision in the slicing framework, which are formulated to minimize the weighted hops for gaining contents. J. Liu *et al.* [19] proposed a hierarchical ICN slice system and determined the quantity, the types, and the locations of the ICN-related VNF from the perspective of the requirements and distribution of ICN users by using the integer linear programming (ILP) method. However, these approaches imply a running cost by reserving resources that might never be used. Moreover, in the 5G Network, MEC has made an outstanding contribution in providing caching, implemented by the User Plane Function (UPF) sinking to the edge of the network. Therefore, dynamic MEC-oriented ICN can be implemented in the 5G architecture driven by the network slicing framework and can improve the network transmission efficiency, network performance and significantly reduce the capital expenditures (CAPEX) and operational expenditures (OPEX) [20] of the ICN Communication Service Provider (ICSP). T. Subramanya *et al.* [21] integrated the MEC node and UPF element to reap the benefits of the MEC system and used the ILP technique to solve the VNF-related dynamic deployment problem. This method could reduce the latency from users to the MEC nodes while ignoring the benefits of the operator. Although these researchers have considered the dynamic deployment and hierarchical caching in ICN slices, there is a lack of comprehensive analysis between the service demand of ICN-UE and the ICSP's revenue in the 5G networks.

By comparing with the existing articles, this paper studies the dynamic hierarchical caching resource allocation problem for the 5G-ICN slice to meet the latency demand of ICN-UE and improve the income of the ICSP as much as possible. To this end, our work has made the following contributions:

1. According to the 5G core network design tenet, we discuss the 5G core network architecture supporting ICN. On this basis, we propose a dynamic hierarchical caching architecture.

2. We comprehensively set up a revenue maximization problem joint caching cost and energy consumption based on the dynamic hierarchical caching architecture. Moreover, we prove the feasibility of the problem and propose an optimal stopping theory (OST)-based dynamic hierarchical caching resource allocation (ODH-CRA) algorithm to solve this problem.

3. We used the ILP method to determine the optimal number and location of the MEC nodes. Based on the latency requirement of the ICN-UE, we used the OST algorithm to find the best redeployment time for the MECs.

4. We propose an improved elite genetic algorithm (IEGA) to solve the hierarchical allocation problem of the internal caching resources of the operator, which can seek a set of optimal solutions to maximize the benefits of the ICSP.

5. Compared with other algorithms, the proposed algorithm can improve the revenue of the ICSP and reduce the latency for users' requesting. Therefore, this scheme provides an efficient caching resource method for the IoT network.

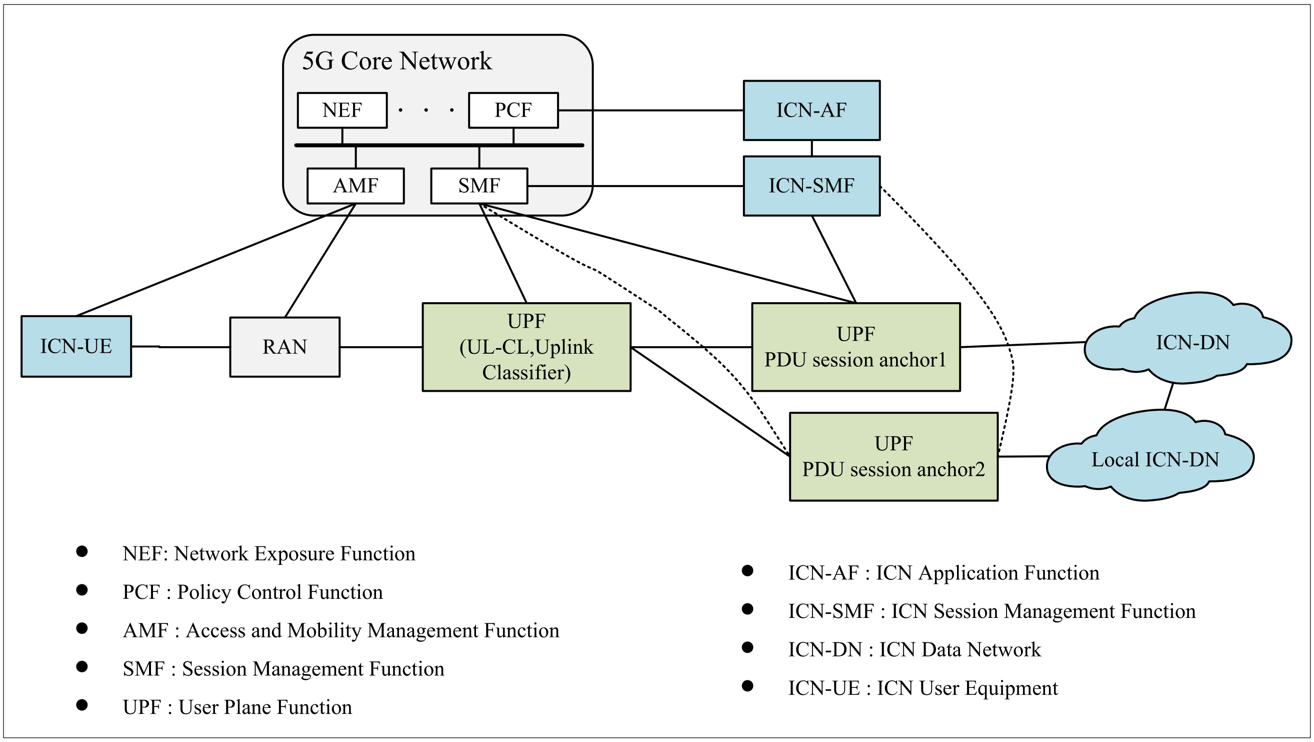
5G Core Network
NEF
PCF
AMF
SMF
ICN-AF
ICN-SMF
ICN-UE
RAN
UPF (UL-CL,Uplink Classifier)
UPF PDU session anchor1
UPF PDU session anchor2
ICN-DN
Local ICN-DN
NEF: Network Exposure Function
PCF : Policy Control Function
AMF : Access and Mobility Management Function
SMF : Session Management Function
UPF : User Plane Function
ICN-AF : ICN Application Function
ICN-SMF : ICN Session Management Function
ICN-DN : ICN Data Network
ICN-UE : ICN User Equipment

FIGURE 1. **5G core network architecture supporting ICN.**

The rest of this article is organized as follows. In Section II, we discuss the network architecture and system model. A dynamic deployment algorithm for the MECs is designed in Section III, including the optimal deployment number, location, and redeployment time of MEC nodes. Section IV introduces an IEGA algorithm to solve the hierarchical caching resource allocation problem within the operator. The simulation results are presented in Section V. In Section VI, we summarize this paper.

II. NETWORK ARCHITECTURE AND SYSTEM MODEL

This section describes the network architecture and system model. We also elaborate on the dynamic deployment of MEC, the expected revenue maximization function, and revenue maximization analysis in detail.

A. NETWORK ARCHITECTURE

Fig. 1 shows the 5G core network architecture supporting ICN [22]. ICN can provide a location-independent cache function, which matches the information-centric nature of IoT applications. In the 5G networks, the ultra-low-latency services are accompanied by the critical cost requirement, which can be solved by placing storage resources with advanced applications to the network edge. Moreover, the core network has formed a cloud interconnection network architecture with the new core and the MECs. On this basis, we propose the dynamic hierarchical caching architecture shown in Fig. 2, which consists of the ICN components such as ICN-Gateway (ICN-GW) and local ICN-DN consisting of the MECs. The ICN components should be implemented by UPF as a user plane function. UPF is used as the protocol data unit (PDU) session anchor point and uplink classifier (UL-CL) in the diagram.

The 5G core network architecture supporting ICN mainly includes the essential functions of the 5G core network and ICN service expansion functions. The ICN Session Management Function (ICN-SMF) is an extension of the Session Management Function (SMF) and is responsible for managing session requests for ICN services. The ICN Application Function (ICN-AF) is an extension of the Application Function (AF) and is responsible for providing business to users. The ICN-AF northbound interface interacts with the 5G operators' Network Exposure Function (NEF) to deploy ICN services and direct traffic. The Access and Mobility Management Function (AMF) reports user location information to the Policy Control Function (PCF) through SMF. According to the user location information and subscription information, PCF adds edge UPF anchor points and inserts UL-CL. When ICN-UE accesses the network, the ICN-SMF communicates with the SMF for PDU sessions and is responsible for flexible forwarding within the ICN-GW and MEC. The SMF is accountable for managing the communication links. As shown in Fig. 2, ICN-UEs set as the red color move to another region of the MEC, resulting in a reassignment of the MEC. When the MEC region can not meet a user's request, the network can trigger the UL-CL function to prevent all users of the MEC region from occupying edge resources.

B. SYSTEM MODEL

The system model includes the ICSP and ICN Communication Service Consumer (ICN-CSC). ICSP is primarily

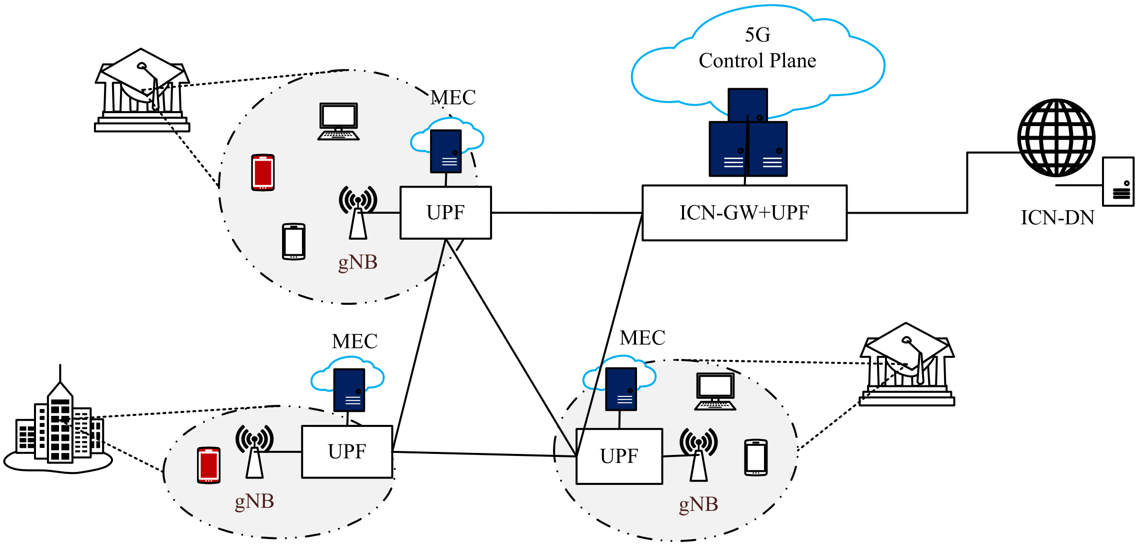

FIGURE 2. Dynamic hierarchical caching architecture.

responsible for providing ICN services, including the design, construction, and operation of ICN services. ICN-CSC is a consumer of ICN services, including the CPs and the ICN-UE. The difference is that CP is responsible for providing content to ICN-UE, and ICN-UE requests content from the ICSP. This article considers only the case where ICN-UE moved within a single operator.

To reap the benefits of ICN and the MECs, we consider a caching architecture consisting of ICN-GW and the MECs implemented by the UPF in the 5G-ICN slice. The ICSP is responsible for managing the caching resource of the 5G-ICN slice. Moreover, the ICSP can allocate the caching resource to the CPs according to the requirements of the ICN-UEs. The CPs can place the content to the cache nodes based on the content demands of the ICN-UEs. However, the request of the ICN-UE can not be satisfied, the dynamic deployment of cache nodes and allocation of the caching resource are considered. The dynamic deployment of the MECs is introduced in subsection C, and resource allocation of the cache nodes is introduced in subsection D and E. At t, we express the number of the MEC as M_{ald}^t, and the number of CP is N. The notations used in this article are summarized in Table 1.

TABLE 1. Table of notations.

Notation	Definition
N	The number of CPs
M_{ald}^t	The number of MECs at t
M	The set of all MECs
s_j	The storage capacity of MEC_j
M_a	The set of MEC potential deployment nodes
M_c	The set of MEC content cache nodes
M_{tar}	The set of MEC target nodes
t_{ac}	The time to request the content
PD_u	The set of PDU sessions requests
k_n	The amount of caching resources requires by CP_n
p	The price of unit caching resources set by the ICSP
α_n	The cost coefficient of CP_n
a_n	The unit cost of caching resources at MEC
a_{CN}	The unit cost of caching resources at the ICN-GW
ω_a	Unit energy comsumption of caching resources to MEC
ω_b	Unit energy comsumption of caching resources to ICN-GW
β	Distribution index of Zipf
$x_{n,j}$	1 if CP_n select MEC_j
$x_{n,CN}$	1 if CP_n select ICN-GW
m_a^{dep}	1 if ICSP needs to deploy MEC in node a
a_a^u	1 if user u's PDU session request is assigned to node a
θ	Service latency requirement of PDU session
Θ	Latency tolerance

C. MEC DEPLOYMENT

To satisfy the demand of the ICSP and reduce the latency for reaching the content, we propose a deployment problem of the MECs in the 5G-ICN slice. The deployment of the MECs is implemented by the UPF, which is responsible for processing data plane packets to the MECs.

The network is represented as a tuple $G(M, L, U)$, where M, L and U denote the sets of all MEC nodes, links between the MECs, and the users with active PDU sessions. The MEC nodes contain the potential deployment nodes M_a, the content cache node M_c and the target node M_{tar}. PD_u denotes the set of PDU session requests. The PDU sessions are characterized by the latency requirement (θ) and the minimum number of the MECs (M_a^u) to guarantee the service quality. The detail will be introduced in Section III.

D. REVENUE MAXIMIZATION FUNCTION

In order to improve the operator's revenue, we formulate a Stackelberg game with the ICSP as leader and CPs as followers for caching resources allocation. The Stackelberg game can maximize the profit function of the player. For convenience, the number of MECs deployed at time t is M_{ald}^t. The ICSP sets the price p per unit of cached resources, and CPs determine the amount of cached resources k_n to purchase from the ICSP based on the ICSP's price p.

We first analyze the profit function of the followers. If the CP purchases caching resources from the ICSP to place a large amount of popular content, the user requests time is reduced, and each CP will obtain revenue by caching the popular content. At the same time, CP needs to pay the rent for purchasing caching resources from the ICSP. The more caching resources purchased, the higher the cost to the CPs. α_n is CP_n's cost coefficient. We can estimate the CP revenue by the following equation:

$$R_n = \log_2\left(1 + \frac{l_n}{\sum_{n'\neq n} l_n}\right) - \alpha_n p k_n \tag{1}$$

$$\alpha_n \geq 0 \tag{1.1}$$

$$\sum_n x_{n,j} k_n \leq s_j \tag{1.2}$$

Constraint (1.1) states that CP_n will pay attention to caching cost. Constraint (1.2) ensures the number of caching resources that all CP_n place cannot exceed the total storage capacity, and s_j is the storage capacity of MEC j.

The probability of the occurrence of any content ranked k obeys the Zipf distribution, denoted as $p(k) = z/k^\beta$, z is a constant, β is the Zipf distribution index. Inspired by [23], for Zipf-like distributions, the cumulative requested probability of the top k_n popular contents can be presented as:

$$l_n = \frac{\sum_{k=1}^{k_n} k^{-\beta}}{\sum_k k^{-\beta}} \approx \frac{k_n^{1-\beta}}{(1-\beta)\Omega} \tag{2}$$

Because the sum interval is limited, we can use the calculus method to get the approximate solution.

The ICSP makes profits by selling caching resources to CPs, and the profit is affected by the price and the amount of caching resources. We define the ICSP's revenue as:

$$R_{ICSP} = \sum_n \left(pk_n - \sum_{j+1} \text{cost}\,(k_n)\right) \tag{3}$$

The cost of ICSP is affected by the caching cost and transmission energy consumption, set the weight factor λ_1 and λ_2 to be equal, and then adjust them according to the requirements of the ICSP. Therefore, $cost(k_n)$ is defined as:

$$\begin{aligned}\text{cost}\,(k_n) = {} & x_{n,j} k_{n,j} \left(\lambda_1 a_n + \lambda_2 w_a\right) \\ & + x_{n,C\,N} k_{n,CN} \left(\lambda_1 a_{CN} + \lambda_2 w_b\right)\end{aligned} \tag{4}$$

$$p \geq 0 \tag{4.1}$$

$$\sum_j x_{n,j} + x_{n,C\,N} = 1 \tag{4.2}$$

$$\lambda_1 + \lambda_2 = 1 \tag{4.3}$$

where a_n is the cost of caching a single resource for the ICSP in the MEC. a_n is determined by the cost of dynamic deployment of the MEC. a_{CN} is the cost of caching a single resource for the ICSP in the ICN-GW. w_a is the energy consumption that the ICSP transfer a single resource from the CP to the MEC. w_b is the energy consumption that the ICSP transfer a single resource from the CP to the ICN-GW. $x_{n,j}(x_{n,j} \in (0, 1))$ and $x_{n,CN}(x_{n,CN} \in (0, 1))$ represent whether the caching resources of CP_n will select the MEC j and the ICN-GW, respectively (=1, yes; =0, no).

Constraint (4.1) states that the price is non-negative. Constraint (4.2) ensures that CP_n can only select one node for content caching at time t. Constraint (4.3) indicates that we need to balance the relationship between caching cost and transmission energy consumption, and the sum of weight factors for cost and energy consumption is 1.

E. REVENUE MAXIMIZATION ANALYSIS

We need to find the perfect Nash equilibrium of the Stackelberg game. In this paper, given the pricing of the ICSP, the relationship between CPs is a non-cooperative game, and when every rational participant will not have the impulse to change the strategy independently [24], there is a Nash equilibrium solution.

Theorem 1: For a non-cooperative game, if the game satisfies: 1) the set of players in the game is limited; 2) The set of strategy space of the game belongs to the bounded closed set in Euclidean space; 3) If the profit function of the non-cooperative game is continuous and concave in the strategy space, there is a Nash equilibrium solution.

Proof: Given the price of caching resources, the number of game players is limited, and the first partial derivative of the function in Eq.(1) is:

$$\frac{\partial R_n}{\partial k_n} = -\alpha_n p - (\beta - 1)\frac{\sum_{n'\neq n} k_{n'}^{1-\beta}}{\sum_{n'\neq n} k_{n'}^{1-\beta} k_n^{1-\beta} + 1} k_n^{-\beta} \tag{5}$$

The second partial derivative of Eq.(1) can be calculated as:

$$\frac{\partial^2 R_n}{\partial\,(k_n)^2} = -c(1-\beta)\frac{\beta k_n^{1-\beta} + c}{\left(ck_n + k_n^\beta\right)^2} \tag{6}$$

where $c = \sum_{n'\neq n} k_{n'}^{1-\beta}$, it is the sum of other CPs' popularity. When $\alpha_n > 0, \lambda_1, \lambda_2 \geq 0, a_n, a_{CN} > 0, x_{i,j}, x_{i,C\,N} \in [0, 1], 0 < \beta < 1, p > 0, k_n > 0$, the set of the strategy space of each player is a bounded closed set in Euclidean space, the profit function is continuous on its strategy space, and the functions of Eq.(1) and Eq.(3) about k_n and p are quasi-concave, so there is a Nash equilibrium solution to the game.

Next, we analyze the ICSP's revenue function of Eq.(3), which can be rewritten as Eq.(7), as shown at the bottom of the next page. The second derivative of the ICSP's revenue function can be calculated as Eq.(8), as shown at the bottom of the next page. Therefore, the Stackelberg equilibrium exists.

Theorem 2: When the second derivative is a strictly concave function in the domain of definition, and the first derivative is a monotone function in the domain, the solution of Nash equilibrium exists and is unique.

Proof: When $0 < \beta < 1, \alpha_n > 0, k_n > 0, \frac{\partial^2 R_n}{\partial (k_n)^2} < 0$. Eq.(6) is a strictly concave function, and the first derivative is monotonically decreasing. We can get from Eq.(9) and Eq.(10), there is a unique extreme value, which is proved by Theorem 2.

$$\lim_{k_n \to 0} \frac{\partial R_n}{\partial k_n} > 0 \tag{9}$$

$$\lim_{k_n \to \infty} \frac{\partial R_n}{\partial k_n} < 0 \tag{10}$$

We have proved that the Stackelberg game exists a Nash equilibrium. The CPs maximize revenue by calculating the optimal amount of caching resources to purchase according to

Eq.(5). The ICSP maximizes revenue by deciding the price of caching resources and choosing the node to allocate caching resources. The problem of maximizing the ICSP's revenue is an NP-hard problem since $p > 0, x_{n,j} \in (0,1)$, and $x_{n,CN} \in (0,1)$.

In Section IV, we propose an IEGA algorithm to solve the optimal caching resources allocation scheme within the ICSP. Algorithm IEGA initializes the price and then evaluates where to allocate the caching resources based on the price. To maximize the revenue of the ICSP, we optimize the scheme x and then search for the price p. The iterative process continues until the change of resource allocation scheme does not follow the price evolution within a predetermined threshold or the predefined maximum number of iterations. After the Stackelberg game reaches the equilibrium, the CPs obtain the optimal amount of caching resources to purchase via Eq.(5).

III. DYNAMIC DEPLOYMENT OF THE MEC

In order to solve the deployment problem of the MECs, we use ILP for solving in this section. This approach considers the deployment cost and service requirements comprehensively. Then, we propose an OST algorithm to reduce latency further.

A. OPTIMAL DEPLOYMENT OF MEC

NFV technology enables the separation of different functions on the same infrastructure [25]. NFV technology provides technical support for the deployment of the MEC. We use ILP method to get the optimal number M_{ald}^{t}. The following is a detailed introduction.

The essential cost of deploying a MEC in a network consists of two parts: deployment cost D_p^{dep} and operation cost D_p^{run}. $m_a^{dep} \in (0,1)$ indicates whether the ICSP needs to deploy MEC when ICN-UE is moving into the MEC region. We can denote C_{dep} as:

$$C_{dep} = \sum_{a \in M} \left(D_p^{dep} + D_p^{run} \right) \cdot m_a^{dep} \tag{11}$$

ICN mainly focuses on the time spent by users requesting content, which is the critical technical indicator to evaluate the performance of ICN. The routing cost of ICN-UE request to content is C_{rou} , $a_a^u \in (0,1)$ states whether user u's PDU session request is assigned to node a. We can denotes C_{rou} as follows:

$$C_{rou} = \sum_{a \in M_a} \sum_{u \in PD_u} \sum_{l \in L} t_{ac}^{u} \cdot a_a^u \tag{12}$$

After MEC is deployed, ICN-UE sends name-based PDU session requests to a MEC node. If the content is found in Content Store (CS) of the MEC node, the content will be sent to the user, and the request time of ICN-UE is t_{cs}. If the content is not found, then the MEC node will check its Pending Interest Table (PIT) for an entry of the same content. If an interest already exists, the ingress interface is added to the existing entry and updates its traffic path to M_c [26]. At the same time, the current interest packet is dropped. The request time for ICN-UE is t_{PIT}^{l}. Otherwise, a new entry is created in PIT. Then, turn to the Forward Information Base (FIB) and forward the interest to the next-hop MEC [27] based on the name prefix. When the packet arrives, MEC looks in the PIT for a match. If found, the MEC will transmit and cache the data to the requesting node. Otherwise, UPF will act as UL-CL and send the request to the core network node ICN-GW, and the request time of ICN-UE is t_{FIB}^{l}.

t_{proc} is the time used by the MEC node to process the PDU session; t_{prop} is the time used by reaching the content, including t_{cs}, t_{PIT}^{l} and t_{FIB}^{l}. The request time to the content t_{ac} consists of t_{proc} and t_{prop}.

$$\begin{aligned} t_{ac} &= t_{proc} + t_{prop} \\ &= t_{proc} + \left(t_{cs} + t_{PIT}^{l} + t_{FIB}^{l} \right) \cdot x_{n,j} \end{aligned} \tag{13}$$

$$t_{ac} \leq \theta \tag{13.1}$$

$$x_{n,j} \in \mathrm{X} \tag{13.2}$$

The constraint (13.1) states that the request time cannot exceed the service time, θ is the service latency requirement of the PDU session. Constraint (13.2) states that the content must be cached where the content is requested.

The ICSP is responsible for the cost of reassignment of the MEC when the ICN-UE moves. The cost is expressed as $C_{a,tar}^{u}$. The reassignment of a PDU session is indicated by $\left[a_{tar}^{u} - a_a^u\right]^{+}$. This expression is 1 if the PDU session has been reassigned. Otherwise, $\left[a_{tar}^{u} - a_a^u\right]^{+}$ is 0.

$$C_{rea} = \sum_{a \in M} \sum_{tar \in M} \sum_{u \in PD_u} C_{a,tar}^{u} \cdot \left[a_{tar}^{u} - a_a^u\right]^{+} \tag{14}$$

The ultimate goal of the MEC deployment is to minimize the ICSP's costs while meeting user latency requirements:

$$\min \left(\eta_1 C_{\text{dep}} + \eta_2 C_{\text{rou}} + \eta_3 C_{\text{rea}} \right) \tag{15}$$

$$\eta_1 + \eta_2 + \eta_3 = 1 \tag{15.1}$$

$$m_a^{\text{dep}} \leq \sum_{u \in U} a_a^u \tag{15.2}$$

$$\begin{aligned} R_{ICSP} &= \sum_{n} \left(pk_n - \sum_{j} x_{n,j} k_{n,j} \left(\lambda_1 a_n + \lambda_2 w_a \right) - x_{n,C\ N} k_{n,CN} \left(\lambda_1 a_{CN} + \lambda_2 w_b \right) \right) \\ &= \sum_{n} \left\{ \left(\frac{1}{\alpha_n p} (1-\beta) \right)^{\frac{1}{\beta+1}} \left(\begin{array}{c} \sum_j x_{n,j} \left(p - \lambda_1 a_n - \lambda_2 w_a \right) \\ + x_{n,CN} \left(p - \lambda_1 a_{CN} - \lambda_2 w_b \right) \end{array} \right) \right\} \end{aligned} \tag{7}$$

$$\frac{\partial^2 R_{ICSP}}{\partial p^2} = -\frac{1}{\alpha_n p^3 (\beta+1)} \left\{ \sum_{n} 2(1-\beta) \left(\sum_j x_{n,j} \left(\lambda_1 a_n + \lambda_2 w_a \right) + x_{n,C\ N} \left(\lambda_1 a_{CN} + \lambda_2 w_b \right) \right) (p+1) \right\} < 0 \tag{8}$$

$$\sum_{a \in M} m_a^{\text{dep}} \leq |M| \tag{15.3}$$

$$\sum_{a \in M} a_a^u \geq M_a^u \tag{15.4}$$

$$\sum_{a \in M} \sum_{tar \in M} \left(a_{tar}^u + a_a^u\right) = 1 \tag{15.5}$$

η_1, η_2, η_3 are the weights of essential cost, routing cost, and reassignment cost in the MEC deployment process, respectively. Constraint (15.1) ensures that we need to comprehensively analyze the importance of deployment cost, routing time, and reassignment cost. Constraint (15.2) indicates that the allocation of PDU sessions during MEC deployment cannot be empty. Constraint (15.3) states that the number of MEC deployments cannot exceed the total number of nodes. Constraint (15.4) ensures that the reliable transmission of a PDU session should be guaranteed, M_a^u is the minimum number of MEC nodes required by a PDU session. Constraint (15.5) indicates that PDU sessions are allocated to at most one MEC node after migration.

B. OPTIMAL STOPPING THEORY

To further reduce the latency of users' requests, we propose an OST algorithm. This method can determine the best redeployment time t^* based on satisfying the users' latency requirements.

The optimal stopping theory determines the time of action based on past events to maximize the average reward [28]. The reward is related to time violations. Specifically, we have a sequence of random variables T_t and a sequence of reward functions $f(T_t)$. If stop at time t, we can get a reward function. The optimal stopping theory determines the stopping time T that maximizes the expected reward, usually called the stopping time $T = t$.

For example, we determine the optimal time t^*, the payoff is calculated as $f(T_{t^*})$. At the stop time $t+1$, $f(T_{t^*})$ cannot be less than the payoff $f(T_{t+1})$.

$$t^* = \inf\left\{t \geq 0 : f\left(T_t\right) \geq \mathrm{E}\left[f\left(T_{t+1}\right) \mid \mathrm{F}\left(T_t\right)\right]\right. \tag{16}$$

Due to the constant change of user location, the status of the MEC deployment can be adjusted due to user latency, service interruptions, and additional overhead. To reduce the deployment cost and latency, we define a random variable T_t^u. If the request time of the PDU session exceeds the service latency requirements at time t, T_t^u is 1. Otherwise, T_t^u is 0.

$$T_t^u = \begin{cases} 1 & \text{if } t_{ac}^u \geq \theta \\ 0 & \text{otherwise} \end{cases} \tag{17}$$

At time t, the number of PDU sessions violating the latency requirement can be defined as:

$$T_t = \sum_{u \in PD_u} T_t^u \tag{18}$$

If this threshold is exceeded, the MEC needs to be redeployed, where A is the expected number of affected PDU sessions and incurs the expected cost $\mathrm{E}[\theta]$.

$$\mathrm{E}[\theta] = \sum_{a \in M} \sum_{t \in M} \sum_{u_\theta \in PD_u} A \cdot \left[a_t^{u\theta} - a_a^{u\theta}\right]^+ \tag{19}$$

To reduce the impact of the MEC redeployment, we need to allocate as many PDU session requests to the MEC as possible at time t, and the number that violates service level should not exceed Θ. If the number of PDU sessions violating latency requirements has exceeded the latency tolerance Θ, we will measure whether MEC needs to be redeployed. T_t is the number of violations of delay tolerance at time t. $\kappa \in [0, 1]$ is a weight factor. $\mathrm{E}[\theta]$ is the redeployment cost after exceeding latency tolerance. Therefore, the maximum number of allowed latency violations can be defined as:

$$f\left(T_t\right) = \begin{cases} T_t & \text{if } T_t \leq \Theta \\ -\kappa \mathrm{E}[\theta] & \text{if } T_t \geq \Theta \end{cases} \tag{20}$$

Our target is to find the optimal time t^* to maximize $f(T_t)$ when giving time series T_t. In other words, if the latency violations are not exceeded, find the time t^* to reach the maximum T_t. If the latency violations are exceeded, find the time t^* to maximize $-\kappa \mathrm{E}[\theta]$. We can define it as follows:

$$\sup_{t \geq 0} \mathrm{E}\left[f\left(T_t\right)\right] \tag{21}$$

According to [29], we evaluate the optimal deployment of the MEC at the next moment. Given that $T_0 = 0$, Θ, $T_1, \ldots, T_t$ and the constraint conditions are satisfied, the optimal stopping time t^* can be obtained as:

$$t^* = \inf\left\{t \geq 0 : \sum_{t=0}^{\Theta - T_t} \left(T_t + t\right) P(T = t) \right.$$
$$\left. + \kappa \mathrm{E}[\theta]\left(1 - \sum_{0}^{\Theta - T_t} \left(T_t + t\right) P(T = t)\right) \leq T_t\right\} \tag{22}$$

Proof: Given $T_t \leq \Theta$, the conditional probability of T_{t+1} is:

$$\begin{aligned} &\mathrm{E}\left[f\left(T_{t+1}\right) \mid T_t \leq \Theta\right] \\ &\quad = \mathrm{E}\left[T_{t+1} \mid T_t \leq \Theta, T_{t+1} \leq \Theta\right] P\left(T_{t+1} \leq \Theta\right) \\ &\qquad + \mathrm{E}\left[\boldsymbol{\kappa} \mathrm{E}[\theta] \mid T_t \leq \Theta, T_{t+1} > \Theta\right] P\left(T_{t+1} > \Theta\right) \\ &\quad = \mathrm{E}\left[T_t + T \mid T \leq \Theta - T_t\right] P\left(T \leq \Theta - T_t\right) \\ &\qquad + \mathrm{E}\left[\boldsymbol{\kappa} \mathrm{E}[\theta] \mid T > \Theta - T_t\right] P\left(T > \Theta - T_t\right) \\ &\quad = \sum_{0}^{\Theta - T_t} \left(T_t + t\right) P(T = t) \\ &\qquad + \kappa \mathrm{E}[\theta]\left(1 - \sum_{0}^{\Theta - T_t} \left(T_t + t\right) P(T = t)\right) \end{aligned} \tag{23}$$

The specific algorithm is presented in Alg. 1. In order to verify the effectiveness of the algorithm, we deploy it in a limited time T_t^{li}, the complexity of the OST algorithm is $O(T_t^{li})$.

IV. HIERARCHICAL CACHING RESOURCE ALLOCATION WITHIN THE ICSP

Network edge caching can improve network performance. In order to make rational use of network resources and reduce the load of edge networks, users with low-performance requirements can migrate their demands of caching resources to the ICN-GW. And caching in the ICN-GW and dynamic MECs decreases the session migration costs of the

Algorithm 1 OST Algorithm

```
Input: Node set $M, L, U$, latency tolerance $\Theta$.
Initialize $T_t^u = 0$.
Derive $T_t$ based on $T_t^u$.
for $t = 1, 2, \ldots, T_t^{li}$ do
    $t^* = \inf\{t \geq 0 : f(T_t) \geq \mathrm{E}[f(T_{t+1}) \mid \mathrm{F}(T_t)]$.
    if $T_t \leq \Theta$ then
        Generate latency violations $T_t$
    else
        Generate latency violations $-\kappa \mathrm{E}[\theta]$
    end if
    Find the optimal time $t^*$ to maximize $f(T_t)$, that can
    be defined as $\sup_{t \geq 0} \mathrm{E}[f(T_t)]$
end for
return $t^*$, and $f(T_t)$
```

high-mobility users and significantly satisfies the low-latency applications in the 5G network environment. Moreover, reasonable caching resource allocation can improve the operator's revenue. In this section, hierarchical caching resource allocation can improve the ICSP's revenue through integrating cache costs and energy consumption. However, the problem of allocating caching resources is an NP-hard problem.

The genetic algorithm (GA) draws on Darwin's theory of evolution and Mendel's theory of heredity. GA can be used to solve game problems and 0-1 knapsack problems [30]. It operates directly on the structure object and has better global optimization ability. Compared with other heuristic algorithms, the solution set is closer to the global optimal. Due to the statistical error in the selection process, the canonical genetic algorithm (CGA) may lose the optimal individual and cannot converge to the global optimal. Therefore, an "IEGA" algorithm is proposed in this paper to directly copy the best individuals in the evolutionary process to the next generation. We use cosine similarity to define the crossover probability, which ensures the system is only performed on the chromosomes with low similarity to improve the accuracy of the solution. Our method can find the global optimal solution of caching resource allocation.

Eq.(3) combines the caching cost and transmission energy consumption, and the ICSP's allocation of the caching resources cannot be determined, which belongs to the NP-hard problem. The IEGA algorithm is proposed to solve the problem. It consists of four parts: selection, crossover, mutation and repair.

In this paper, the genetic code is represented as $x_{n,j+1}$, the maximum number of iterations is It, and the population number is I. The chromosomes are denoted by $X_i = (x_{n,1}, x_{n,2}, \ldots, x_{n,M^t_{ald}+1})$. Population is expressed as $X = (X_1, X_2, \ldots, X_I)$. If the corresponding individual cannot find a MEC node to respond to CP's request, the individual should be reinitialized until the condition is met. If a CP is paired with multiple MEC nodes, the node with the most significant profit and satisfying the constraints should be selected. Otherwise, the individual should be reinitialized.

In the IEGA algorithm, the most common selection strategy is the proportional selection strategy. The probability of an individual is the ratio of the difference between individual fitness and minimum fitness to the sum of all individual differences. $f(X_i)$ is individual fitness value, f_{min} is the minimum value of individual fitness, and the probability of X_i is expressed as P_{X_i}, which is the possibility of being selected for offspring breeding. we can define it as follows:

$$P_{X_i} = \frac{f(X_i) - f_{\min}}{\sum_{i=1}^{I}(f(X_i) - f_{\min})} \tag{24}$$

For the ω round, $\xi_w \in U(0,1)$. When satisfied $PP_{X_i-1} < \xi_w < PP_{X_i}$, X_i is selected [31]. Existing work has proved that the elitist genetic algorithm is globally convergent [32].

The purpose of crossover is to produce new offspring by combine two chromosomes. Many scholars recommend selecting a crossover probability in a fixed value. In this paper, we improve the cross probability, which is defined by a cosine probability. Furthermore, its crossover operation can only be manipulated when the two chromosomes have low similarity. This operation is carried out to reduce the probability of producing unnecessary chromosomes [33]. Chromosome X_i and chromosome $X_{i'}$ cosine similarity is defined as:

$$\cos(X_i, X_{i'}) = \frac{X_i \cdot X_{i'}}{|X_i|\,|X_{i'}|} \tag{25}$$

Mutation operation is to maintain the diversity of the population. In this paper, we do not change the mutation probability. The mutation probability is Pm. A repair operation follows the mutation is to remove the chromosomes that do not meet the requirements. Suppose the CP has not yet sent a request to the ICSP corresponding to one or more new individuals. In that case, the individual will be replaced by the corresponding individual from the previous generation population.

We can obtain the ultimate solution of hierarchical caching resources allocation through selection, crossover, mutation, repair operations. IEGA algorithm is shown in Alg. 2. We are assuming that the algorithm converges within it_{st} iterations, where each iteration has the complexity of $O(I^3)$. Therefore, the overall complexity of the IEGA algorithm is $it_{st} \times O(I^3)$. The IEGA algorithm considers the population crossover characteristics, and the computational complexity is relatively higher than the greedy algorithm.

V. PERFORMANCE EVALUATION

A. SIMULATION SETTINGS

For simulations, we used the GT-ITM tool to generate a network slice instance request. We implemented the MEC deployment model using the Python-based package Pyomo [34] and Gurobi as the underlying solver. All simulations were performed on a computer configured with Intel(R) Core(TM) i5-6300HQ CPU@2.30GHZ and 16GB of RAM. In the simulation, eight MECs were considered, and the capacity of each MEC is $s_j = 100$, the capacity of ICN-GW is twice that of the MEC. In the dynamic deployment of MEC, we consider deployment, routing costs,

```
Algorithm 2 IEGA Algorithm
Input: Initialization population $X$, population number $I$,
    multiplication algebra $It$, node set $M^t_{ald}+1$, CP set $N$,
    ICN-MVNO's initial price $p$, storage capacity collection
    $S=(s_1, s_2, \dots, s_{M^t_{ald}+1})$.
Output: Get the optimal solution $X_i^*$.
According to Eq.(1) and Eq.(3) based on the initial price
    $p$, generate $CP_n$'s caching resource request list $k_n$;
$R_{ICSP} \leftarrow 0$.
According to $k_n$, select $CP_n$ and map $M^t_{ald}+1$ and $CP_n$
    as a set of genetic coding.
Determine the fitness $R_{ICSP}$ and feasibility of solutions
    via Eq.(3).
for $it = 1, 2, \dots, It$ do
    Calculate the probablity that each individual $i$ can be
    selected by Eq.(24).
    Select individuals according to probability $P_X$.
    if $R_{ICSP}(X_{i-1}) > R_{ICSP}(X_i)$, $(i \neq i-1)$ then
        $X = X + X_{i-1}$
        According to Eq.(25), calculate similarity matrix.
        Crossover via correlation matrix.
        Mutation via probability $Pm$.
        Repair the set and eliminate the duplicate individ-
    uals.
        Find the optimal solution $X_i^*$.
    end if
    $it = it + 1$
end for
return $result$ $X_i^*$, $R_{ICSP}$
```

TABLE 2. Simulation parameters.

Parameter	Value
The initial number of MEC deployments	3
The number of ICN - GW	1
The number of CP	3
U(each with one active PDU session)	100
β	0.8
α_n	0.01
λ_1, λ_2	$\lambda_1 = 0.3, \lambda_2 = 0.7$
θ	20 ms
a_{CN}	2
ω_a, ω_b	$\omega_a = 8.39, \omega_b = 22.653$

and reassignment and giving more importance to the latter ($\eta_1 = 0.3, \eta_2 = 0.3, \eta_3 = 0.4$) [35]. We model the number of sessions with latency violations as a Poisson distribution with a mean of $\lambda = 20$. The basic simulation parameters are shown in Table 2. We further compared it with latency-aware service placement and live migration (LALM) [36], the best-availability algorithm that the current demand queue has the earliest finish time[37], and the low-latency algorithm that the scheme sets up new VNF for each demand [38].

B. FEASIBILITY ANALYSIS

The simulation parameters of the IEGA algorithm are shown in Table 3. In order to test the feasibility of the IEGA, we compare it with the genetic placement (GP) algorithm [39]. The purpose of the GP algorithm is to find the optimal global solution in the hierarchical caching architecture. The selection method is roulette, with a crossover probability of 0.8 and a mutation probability of 0.06. We provide the optimal solution obtained using a dynamic programming algorithm to the MEC caching problem compared to the proposed IEGA algorithm. As shown in Fig. 3, the results show that the latency in the MEC gradually decreases to a fixed value. The distance between the IEGA algorithm and the optimal solution gradually decreases, and the value between the GP algorithm and the optimal solution shows the same trend. At the beginning of the iteration, the latency of the IEGA algorithm is slightly higher than that of the GP algorithm due to inaccurate crossover probability. When the number of iterations reaches 200, the performance of the two algorithms tends to be stable, and the performance of the IEGA algorithm is better than that of the GP algorithm.

TABLE 3. Simulation parameters of the IEGA algorithm.

Parameter	Value
Population size I	50
The mutation probability	0.01
Number of iterations It	300

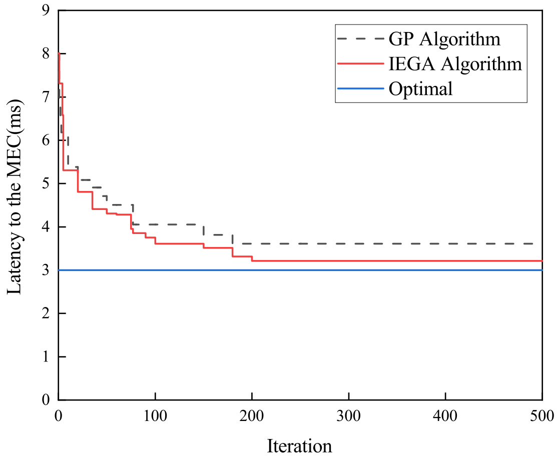

FIGURE 3. Optimisation process.

C. IMPACT OF LATENCY VIOLATIONS

Latency violations may occur when the deployed MEC can no longer meet the user's requirements. However, not all deviations are responsible for latency violations, while deviations from an optimal deployment happen at any time. Because the latency violations can distribute between all users and content sources, it is essential to note that the value of latency violations is the key to finding optimal redeployment time. The number of the latency violations will be shown in Table 4. Eq.(23) can solve the problem of finding redeployment probability at the next moment.

To show the performance of dynamic algorithms, we compare the impact of latency violations on the benefits of the ICSP under different algorithms. The Iterative Greedy and Search (IGS) algorithm [40] and the GP algorithm using the genetic placement method are static deployment algorithms. Fig. 4 illustrates that the number of latency violations affects

TABLE 4. Number of latency violations.

Number of the latency violations	Time slot
0	10
0	15
3	20
1	25
0	30

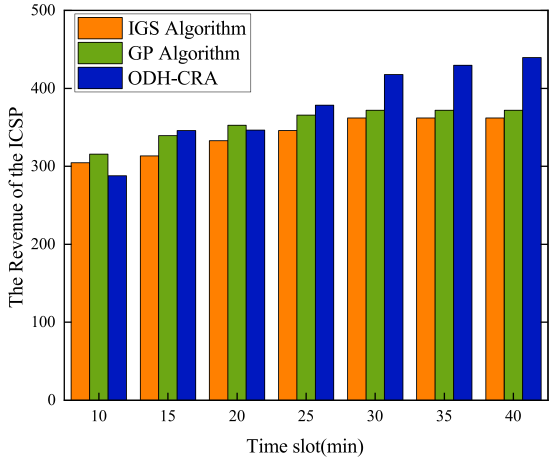

FIGURE 4. Impact of the latency violations on ICSP revenue.

the benefits of the ICSP. It can be seen that when MEC is deployed statically, the ICSP benefits have barely increased in general. With the increase of deployment time, the benefits of the algorithm presented in this paper show a significant increase. At 10min, the ICSP gains were lower than static deployment due to the higher cost of dynamic deployment of MEC. At 20min, due to the violation of the latency tolerance condition, MEC generates additional cost expenditure, resulting in decreased revenue. At 25min, the number of latency violations is 1, and the income of the ICSP increases less. Over the following period, ICSP's revenue of the proposed algorithm is far more than the other algorithm.

D. IMPACT OF THE CACHING ARCHITECTURE

In order to better reflect the advantages of dynamic hierarchical caching architecture, we first test the latency of ICN-UE requests under four conditions at time t: 1) ICN-GW only: the content is only cached in ICN-GW. 2) MEC only case: the content is only cached in MEC. 3) ICN - GW+MEC+ODH-CRA. 4)ICN-GW + MEC situation: the content is cached in ICN-GW and MEC. If the local MEC node is not hit, the content is directly requested to ICN-GW. As shown in Fig. 5, our hierarchical caching architecture significantly reduces latency. For example, when the cache size is 80%, the latency performance of our scheme is 36% improvement over the MEC only scheme, 61% better than that of ICN-GW only scheme, and 24% improvement over the ICN-GWzMEC scheme.

E. LATENCY EVALUATION

Fig. 6 illustrates the impact of cache capacity on the latency performance of different algorithms. With the increase of cache space, the ODH-CRA algorithm showed a decreasing trend. Compared to the LALM and the best-availability, the ODH-CRA can reduce the latency by about 9%, 21%, respectively. The benefits come from the fact that, in the ODH-CRA, an integrating method with the service requirements and latency violations is found to satisfy the frequent service migration requirements in the limited service area. Moreover, the latency of the ODH-CRA algorithm is higher than that of other algorithms. This is because when the cache space is small, we pay more attention to the impact of the ICSP's revenue under the condition of satisfying the users' demands. The latency of the ODH-CRA algorithm is higher than that of the low-latency algorithm, which focuses on latency requirements.

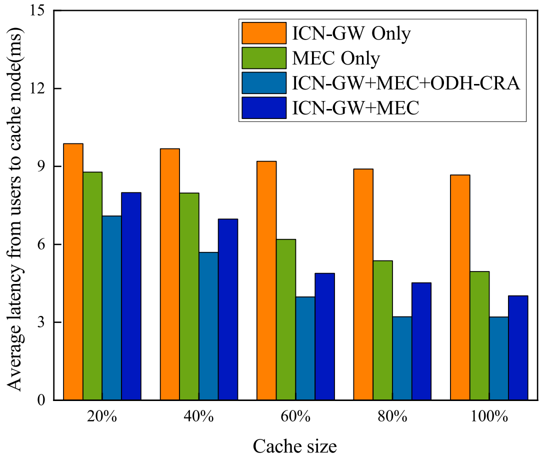

FIGURE 5. Impact of the caching architecture on latency.

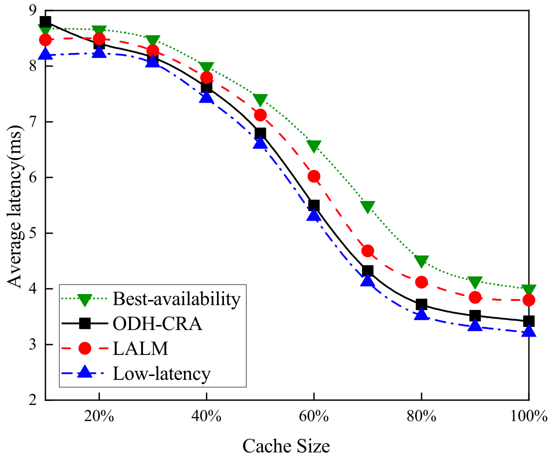

FIGURE 6. Influence of cache size on latency.

Fig. 7 shows the impact of the number of PDU session requests on latency. To avoid the limited cache capacity, we set the capacity of the MEC to a fixed value of 100. The ODH-CRA algorithm determines the best redeployment time through the OST algorithm. When the redeployment time is reached, the ICSP redeploys the MECs to meet the latency requirements of user requests. As shown in Fig. 7, the performance of the ODH-CRA algorithm is significantly better than the LALM algorithm satisfying the latency of ICN-UE

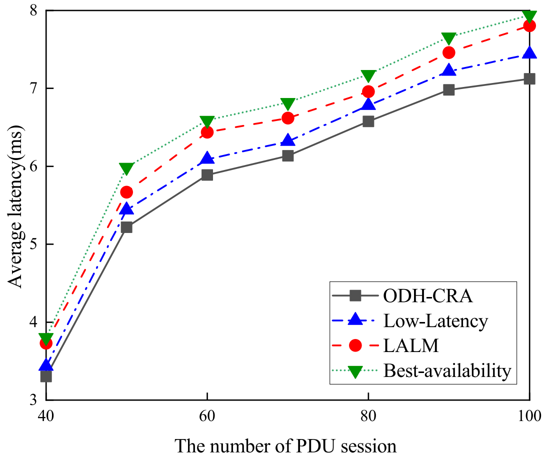

FIGURE 7. Effect of the number of PDU sessions on latency.

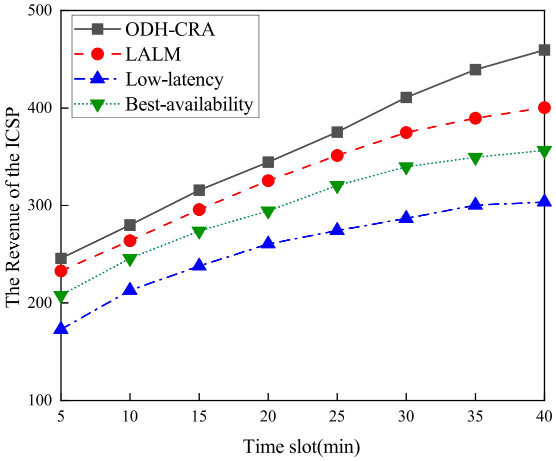

FIGURE 8. Evolution of the revenue.

by increasing the latency of service migration. Moreover, the best-availability and the low-latency algorithm ignoring the number of latency violations of the ICN-UE.

F. REVENUE EVALUATION

Fig. 8 displays the change of the ICSP's revenue under the four algorithms. On the whole, the income of the ICSP is in an upward trend. At first, the ODH-CRA algorithm grew slowly. When the time reached 25min, the growth rate of the ODH-CRA algorithm was significantly higher than that of other algorithms. Compared to the LALM algorithm, the best-availability, low-latency, the ODH-CRA can increase the revenue by about 14%, 28%, 51%, respectively. This is because the algorithm in this paper considers deployment costs, service requirements, and reassignment costs. When the deployment is relatively stable, the ODH-CRA algorithm can maximize the benefits of the ICSP while satisfying the service requirements.

VI. CONCLUSION AND FUTURE WORK

The paper concluded by arguing the dynamic hierarchical characteristics of nodes when studying the 5G-ICN slice caching resource allocation. First, we built a dynamic hierarchical caching architecture. On this basis, we set up a revenue maximization problem under the condition that users' service requirements cannot be satisfied. To solve this problem, we proposed an ILP approach that combines deployment cost and latency requirements to find the optimal deployment location and number of the MECs. Then, we used the time-category-based OST method to determine the best redeployment time of the MECs to reduce latency further. In addition, to further enhance the benefits of the ICSP, we used the IEGA algorithm to find the optimal solution of caching resources allocation. Finally, we compared our algorithm with the other three approaches and compared the impact of different caching architectures. The proposed algorithm has a significant improvement in the ICSP revenue and user request latency.

With the rapid development of mobile communication networks, the dynamic hierarchical deployment of network cache nodes is inevitable. Our solutions could meet users' needs, improve the ICSP's revenue, and reduce user request latency. Our proposed solution is a step forward in deploying the 5G-ICN slice, which is critical for an effective IoT scheme. In the future, we plan to consider the hierarchical caching problem while joint the cooperation between nodes. Caching cooperation, in this case, are complex, and we need to investigate them further.

REFERENCES

[1] S. Arshad, M. A. Azam, M. H. Rehmani, and J. Loo, "Recent advances in information-centric networking-based Internet of Things (ICN-IoT)," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 2128–2158, Apr. 2019.

[2] S. Liao, J. Wu, J. Li, and K. Konstantin, "Information-centric massive IoT-based ubiquitous connected VR/AR in 6G: A proposed caching consensus approach," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5172–5184, Apr. 2021.

[3] I. U. Din, H. Asmat, and M. Guizani, "A review of information centric network-based Internet of Things: Communication architectures, design issues, and research opportunities," *Multimedia Tools Appl.*, vol. 78, no. 21, pp. 30241–30256, Nov. 2019.

[4] M. Liyanage, P. Porambage, A. Y. Ding, and A. Kalla, "Driving forces for multi-access edge computing (MEC) IoT integration in 5G," *ICT Exp.*, vol. 7, no. 2, pp. 127–137, Jun. 2021.

[5] O. Serhane, K. Yahyaoui, B. Nour, and H. Moungla, "CnS: A cache and split scheme for 5G-enabled ICN networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2020, pp. 1–6.

[6] C. Xu, M. Wang, X. Chen, L. Zhong, and A. L. Grieco, "Optimal information centric caching in 5G device-to-device communications," *IEEE Trans. Mobile Comput.*, vol. 17, no. 9, pp. 2114–2126, Sep. 2018.

[7] C. Liang, F. R. Yu, and X. Zhang, "Information-centric network function virtualization over 5G mobile wireless networks," *IEEE Netw.*, vol. 29, no. 3, pp. 68–74, May 2015.

[8] I. U. Din, S. Hassan, M. K. Khan, M. Guizani, O. Ghazali, and A. Habbal, "Caching in information-centric networking: Strategies, challenges, and future research directions," *IEEE Commun. Surveys Tuts.*, vol. 20, no. 2, pp. 1443–1474, 2nd Quart., 2018.

[9] M. Hajimirsadeghi, N. B. Mandayam, and A. Reznik, "Joint caching and pricing strategies for popular content in information centric networks," *IEEE J. Sel. Areas Commun.*, vol. 35, no. 3, pp. 654–667, Mar. 2017.

[10] Y. Song, C. Yang, J. Shen, and L. Chang, "Joint delay and energy management for cache-enabled ultra-dense cellular networks: A game-theoretic learning," in *Proc. IEEE Int. Conf. Commun. Syst. (ICCS)*, Dec. 2018, pp. 401–406.

[11] F. Shen, K. Hamidouche, E. Bastug, and M. Debbah, "A stackelberg game for incentive proactive caching mechanisms in wireless networks," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–6.

[12] R. Ravindran, P. Suthar, A. Chakraborti, S. O. Amin, A. Azgin, and G. Wang, "Deploying ICN in 3GPP's 5G NextGen core architecture," in *Proc. IEEE 5G World Forum (5GWF)*, Jul. 2018, pp. 26–32.

[13] A. Ndikumana, S. Ullah, T. LeAnh, N. H. Tran, and C. S. Hong, "Collaborative cache allocation and computation offloading in mobile edge computing," in *Proc. 19th Asia–Pacific Netw. Oper. Manage. Symp. (APNOMS)*, Sep. 2017, pp. 366–369.

[14] R. Xie, J. Wu, R. Wang, and T. Huang, "A game theoretic approach for hierarchical caching resource sharing in 5G networks with virtualization," *China Commun.*, vol. 16, no. 7, pp. 32–48, Jul. 2019.

[15] Y. K. Tun, N. H. Tran, D. T. Ngo, S. R. Pandey, Z. Han, and C. S. Hong, "Wireless network slicing: Generalized Kelly mechanism-based resource allocation," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 8, pp. 1794–1807, Aug. 2019.

[16] Z. Zhang, C.-H. Lung, M. St-Hilaire, and I. Lambadaris, "Smart proactive caching: Empower the video delivery for autonomous vehicles in ICN-based networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 7, pp. 7955–7965, Jul. 2020.

[17] R. Ravindran, A. Chakraborti, S. O. Amin, A. Azgin, and G. Wang, "5G-ICN: Delivering ICN services over 5G using network slicing," *IEEE Commun. Mag.*, vol. 55, no. 5, pp. 101–107, May 2017.

[18] H. Jin, H. Lu, Y. Jin, and C. Zhao, "IVCN: Information-centric network slicing optimization based on NFV in fog-enabled RAN," *IEEE Access*, vol. 7, pp. 69667–69686, 2019.

[19] J. Liu, B. Zhao, M. Shao, Q. Yang, and G. Simon, "Provisioning optimization for determining and embedding 5G end-to-end information centric network slice," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 1, pp. 273–285, Mar. 2021.

[20] W. Zhuang, Q. Ye, F. Lyu, N. Cheng, and J. Ren, "SDN/NFV-empowered future IoV with enhanced communication, computing, and caching," *Proc. IEEE*, vol. 108, no. 2, pp. 274–291, Feb. 2020.

[21] T. Subramanya, D. Harutyunyan, and R. Riggio, "Machine learning-driven service function chain placement and scaling in MEC-enabled 5G networks," *Comput. Netw.*, vol. 166, Jan. 2020, Art. no. 106980.

[22] G. Gur, P. Porambage, and M. Liyanage, "Convergence of ICN and MEC for 5G: Opportunities and challenges," *IEEE Commun. Standards Mag.*, vol. 4, no. 4, pp. 64–71, Dec. 2020.

[23] L. Breslau, P. Cao, L. Fan, G. Phillips, and S. Shenker, "Web caching and Zipf-like distributions: Evidence and implications," in *Proc. IEEE INFOCOM Conf. Comput. Commun. 18th Annu. Joint Conf. IEEE Comput. Commun. Societies. Future Now*, Mar. 1999, pp. 126–134.

[24] P.-y. Nie and P.-A. Zhang, "A note on stackelberg games," in *Proc. Chin. Control Decis. Conf.*, Jul. 2008, pp. 1201–1203.

[25] J. Li, J. Wu, G. Xu, J. Li, X. Zheng, and A. Jolfaei, "Integrating NFV and ICN for advanced driver-assistance systems," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 5861–5873, Jul. 2020.

[26] S. Shailendra, S. Sengottuvelan, H. K. Rath, B. Panigrahi, and A. Simha, "Performance evaluation of caching policies in NDN–an ICN architecture," in *Proc. IEEE Region Conf. (TENCON)*, Nov. 2016, pp. 1117–1121.

[27] V. Sivaraman, D. Guha, and B. Sikdar, "Optimal pending interest table size for ICN with mobile producers," *IEEE/ACM Trans. Netw.*, vol. 28, no. 4, pp. 1615–1628, Aug. 2020.

[28] D. Zheng, W. Ge, and J. Zhang, "Distributed opportunistic scheduling for ad hoc networks with random access: An optimal stopping approach," *IEEE Trans. Inf. Theory*, vol. 55, no. 1, pp. 205–222, Jan. 2009.

[29] R. Cziva, C. Anagnostopoulos, and D. P. Pezaros, "Dynamic, latency-optimal vNF placement at the network edge," in *Proc. IEEE INFOCOM Conf. Comput. Commun.*, Apr. 2018, pp. 693–701.

[30] M. Montazeri, R. Kiani, and S. S. Rastkhadiv, "A new approach to the restart genetic algorithm to solve zero-one knapsack problem," in *Proc. IEEE 4th Int. Conf. Knowl.-Based Eng. Innov. (KBEI)*, Dec. 2017, pp. 50–53.

[31] X. Li, Z. Wang, Y. Sun, S. Zhou, Y. Xu, and G. Tan, "Genetic algorithm-based content distribution strategy for F-RAN architectures," *ETRI J.*, vol. 41, no. 3, pp. 348–357, Jun. 2019.

[32] K. Singh and A. S. Pillai, "Schedule length optimization by elite-genetic algorithm using rank based selection for multiprocessor systems," in *Proc. Int. Conf. Embedded Syst. (ICES)*, Jul. 2014, pp. 86–91.

[33] Z. Xue, "Routing optimization of sensor nodes in the Internet of Things based on genetic algorithm," *IEEE Sensors J.*, early access, Mar. 24, 2021, doi: 10.1109/JSEN.2021.3068726.

[34] W. E. Hart, J.-P. Watson, and D. L. Woodruff, "Pyomo: Modeling and solving mathematical programs in Python," *Math. Program. Comput.*, vol. 3, no. 3, pp. 219–260, 2011.

[35] I. Leyva-Pupo, C. Cervelló-Pastor, C. Anagnostopoulos, and D. P. Pezaros, "Dynamic scheduling and optimal reconfiguration of UPF placement in 5G networks," in *Proc. 23rd Int. ACM Conf. Modeling, Anal. Simulation Wireless Mobile Syst.*, Nov. 2020, pp. 103–111.

[36] B. E. Mada, M. Bagaa, T. Tale, and H. Flinck, "Latency-aware service placement and live migrations in 5G and beyond mobile systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2020, pp. 1–6.

[37] R. Mijumbi, J. Serrat, J.-L. Gorricho, N. Bouten, F. De Turck, and S. Davy, "Design and evaluation of algorithms for mapping and scheduling of virtual network functions," in *Proc. 1st IEEE Conf. Netw. Softwarization (NetSoft)*, Apr. 2015, pp. 1–9.

[38] K. Yang, H. Zhang, and P. Hong, "Energy-aware service function placement for service function chaining in data centers," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–6.

[39] Q. Tang, R. Xie, T. Huang, and Y. Liu, "Hierarchical collaborative caching in 5G networks," *IET Commun.*, vol. 12, no. 18, pp. 2357–2365, Nov. 2018.

[40] Y. Fan, L. Wang, W. Wu, and D. Du, "Cloud/Edge computing resource allocation and pricing for mobile blockchain: An iterative greedy and search approach," *IEEE Trans. Comput. Social Syst.*, vol. 8, no. 2, pp. 451–463, Apr. 2021.

LIPING GE received the B.S. degree in communication engineering from Cangzhou Normal University, Cangzhou, China, in 2019. She is currently pursuing the M.S. degree in communications engineering with Beijing Information Science and Technology University, Beijing, China. Her research interests include network slicing, network caching, and game theory.

JINHE ZHOU received the B.S. and M.S. degrees in radio physics from Wuhan University, Hubei, China, in 1988 and 1991, respectively. He is currently a Professor with the School of Information and Communication Engineering, Beijing Information Science and Technology University. He has been the author of more than 50 articles. He has hosted and participated in several scientific research projects, including the National Key Project of Hi-Tech Research and Development Program of China (973 Program) and the National Natural Science Foundation of China. His research interests include 5G networks, edge computing, game theory, and green information-centric networks. He was a recipient of Beijing Famous Teacher Award.

ZHENKANG ZHENG received the B.S. degree in electronic science and technology from Wuhan University of Technology, Wuhan, China, in 2020. He is currently pursuing the M.S. degree in communications engineering with Beijing Information Science and Technology University, Beijing, China. His research interests include network slicing, cache technology, and resource allocation.