EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

Content-oriented network slicing optimization based on cache-enabled hybrid radio access network

Hao Jin*, Haiya Lu and Chenglin Zhao

Abstract

With the development of smart mobile devices and various mobile applications, content-oriented service has become the most popular service which occupies network resources and results in high traffic load. In order to improve quality of experience in radio access network and reduce the OpEx and CapEx of operators, wireless network virtualization and network slicing come into the vision and are deemed as promising solutions to radio access networks to provide tailored services. Therefore, network slicing and optimization based on content-oriented service become a challenging research direction. In this paper, network slicing and resource optimization on content-oriented application in cache-enabled hybrid radio access network based on complex network are investigated. A Cooperative Network Slicing Framework Based on Content in RAN (CNSC-RAN) is presented. Based on CNSC-RAN, procedures of content-oriented static network slicing and dynamic slicing are proposed. Content-oriented slicing is modeled and analyzed which includes slicing on content cache resources and communication resources. In order to obtain the optimized resources sliced for each content, the optimization problem is formulated to minimize the average system cost to get the contents required by users. The problem is solved by a heuristic algorithm called CCSOA (Content-Centric Slicing Optimization Algorithm) in a dynamic content-oriented network slicing procedure enabling UEs with self-evicting contents. The performance of CCSOA is evaluated by performance metrics including hit rate, average cache occupation, average system cost, and request traffic reduction to macro cell base station comparing with CEE and ProbCache. Simulation results reveal the effectiveness of CCSOA.

Keywords: Content caching, Network slicing, Complex network, Caching optimization

1 Introduction

With the development of smart mobile devices and various mobile applications, content-oriented service has become the most popular service which occupies network resources and results in high traffic load. Since users are usually interested in the contents themselves than where the contents are located, information-centric networking (ICN) is proposed as a new paradigm. In ICN, mobile users publish the information-centric contents by uploading their contents to the cache-enabled routers in the network, and at the same time, they require for the information-centric contents and get the contents they required from ICN by downloading the contents caching from the routers. In order to reduce service delay and improve hit rate for contents required by users and the efficiency of communication resources, caches are deployed at the edge of ICN, which motivates the deployment of caches in heterogeneous radio access network including macro cell base station (MBS), small cell base station (SBS), and even in the user devices, fog radio access network(F-RAN) is proposed as an efficient solution which can achieve high spectral efficiency/energy efficiency, low latency, and fantastic reliability for tailored services in 5G. The rapid and affordable scaling that make F-RANs adaptive to the dynamic traffic and radio environment, and low burdens on the fronthaul and the base band unit(BBU) pool [1], which becomes a perspective way to improve QoE (quality of experience)

* Correspondence: hjin@bupt.edu.cn
The Key Laboratory of Universal Wireless Communications for Ministry of Education, Beijing University of Posts and Telecommunications, Beijing 100876, China

for different services and reduce OpEx and CapEx of operators in radio access network.

In order to achieve optimal performances in F-RAN, wireless network virtualization (WNV), and network slicing are proposed as promising solutions to heterogeneous radio access networks. Network function virtualization (NFV) management and orchestration (MANO) supports a wide range of services by orchestrating the VNF (virtual network function) deployment and operation across diverse computing, caching, and networking resources [2–4]. According to VNF, WNV creates logical partitions of some existent physical wireless network functions and resources in an efficient manner and allows network configuration to be tailored for different scenarios and applications [2–6].

The key technologies of WNV include wireless network function virtualization and wireless resource virtualization. Wireless network function virtualization leads to various deployment solutions such as cloud radio access network(C-RAN) and F-RAN. The motivations for virtualizing a wireless network diverse from enabling the infrastructure sharing among several operators, offering a layer of abstraction to simplify network management and programmability of wireless networks as well as network slicing based on service, user, or application [6–8]. Virtualization of a wireless network can be applied at different layers and degrees. Virtualizing radio spectrum and physical layer of base stations allows sharing spectrum as well as common access network infrastructure, which includes opportunistic sensing to discover scenario-dependent optimal platform [9], deep sensing for spectrum resources [10, 11], and radio resources allocation in RAN. Another respect of virtualization resources is computing, storage, and networking resources deployed in RAN regarding the requirements of cloudified applications [6–8].

Logical partitions in NFV is also known as resource slicing, which divides the network into slices made of different resources and capacities so as to offer differentiated services for heterogeneous use cases and enable creation of customized services with fine control features of QoS [7, 8, 12, 13]. Orchestration at the operator level can be seen as a special instance of the virtual network embedding (VNE) problem, in which the combined network slicing optimization problem is to make decision on optimal placement of VNFs at resource nodes, jointly the necessary link capacity reservations for their interconnection, under additive link and node capacity constraints so as to minimize the overall resource utilization cost [14]. The typical slicing motivation are different operators, different infrastructures and resource level, as well as various service requirements, including low power, low-data-rate machine-type communication, high data rate multimedia, and delay-sensitive applications [8]. Among the network slicing issues, network slicing based on content-oriented service become a challenging research direction as one of the important services provided by heterogeneous radio access network [7, 14].

Research issues on wireless network slicing based on content-oriented service mainly focus on network virtualization architecture and optimization of virtual wireless network resources. In the respect of network virtualization architecture based on caching, the proposed architectures include service based [12, 13, 15–17], application based [13, 18–20], MVNO (mobile virtual network operator) based [13, 16, 20, 21], access network [13, 15] and/or core network based [22], and different physical layer resource based [8]. In the respect of optimization of virtual wireless network resources based on content-oriented caching, since slicing on a wireless network involves how to assign resources to different slices, which can be transformed into the optimization problem for various optimization objectives, the resources of network slicing can be the available radio spectrum (divided also by time, frequency, or space), wireless network infrastructures, the available transmission time, or the capacity of the medium [8], while content-oriented optimization objects diverse in network-centric objectives and user-centric objectives, and the optimization also depends on the network architecture [23], the state-of-the-art research issues on optimization of virtual wireless network resources based on content caching are investigated, and several research issues concentrate on network slicing optimization integrating network function virtualization and in-network caching.

Typical issues are concentrated on optimization for MVNOs [17, 24–27]. For example, in [17], an information-centric wireless network virtualization architecture is proposed including radio spectrum resource, wireless network infrastructure, virtual resources (including content-level slicing, network-level slicing, and flow-level slicing), and information-centric wireless virtualization controller. The virtual resource allocation and in-network caching strategy is formulated as an optimization problem considering the gain of virtualization and in-network caching. In [24], the optimization problem is formulated as minimizing the average delay based on multi-object auctions in small-cell networks, while in [25], the optimization objective is to minimize the inter-MNO (mobile network operator) and intra-MNO traffic load based on a cloud-based virtual mobile caching framework; the virtual resource allocation and caching strategies are formulated as a joint optimization problem considering the gain of not only virtualization but also caching in the proposed information-centric wireless network virtualization architecture with D2D (device to device) communications [26]. In [27], a network slicing framework for multi-tenant heterogeneous cloud radio access networks (H-CRAN) is presented and the network slicing process is formulated as a weighted throughput maximization problem that involves sharing of computational resources, fronthaul capacity, physical remote radio heads, and radio resources in the single-cell scenario.

Slicing of physical layer resources in RAN (radio access network) is focused in [28, 29]. In [28], a resource slicing framework for radio access networks considering D2D communication pairs is proposed; by formulating the optimization problem as maximizing the total achieved throughput for the deployed links and the upper bounds are given on the achievable performance as well as low-complexity sub-optimal greedy-based algorithms. In [29], a network virtualization substrate is described in virtualizing a base station's uplink and downlink resources into slices in cellular networks. A provably optimal slice scheduler is implemented, which enables existence of slices with bandwidth-based and resource-based reservations simultaneously, and customized flow scheduling within the base station on a per-slice basis.

Some optimization issues are designed from the point of view of VNF placement [14, 30, 31]. In [30], a VNF placement heuristic named wireless network embedding (WiNE) is presented to optimize the number of accepted requests, the average embedding cost, the average node, and link utilization with k–ary fat–tree substrate network. In [31], a recursive algorithm for joint VNF placement and VNF chaining is proposed and evaluated in a in a real NFV enabled cloud environment in order to optimize resource usage, acceptance rate, and provider's revenues.

In [32, 33], joint optimization considering storage, computing, and network resources is investigated. In [32], based on a framework of information-centric HetNet (heterogeneous network) with a MBS and multiple SBSs, the virtual resource allocation was formulated to maximize the aggregate utility of the MVNO system including communication, computation, and caching revenue. In [33], considering the downlink transmission case in a cellular HetNet comprised of cache-enabled MBSs, SBSs, a central SDN (software-defined network) controller, and a content server, a joint optimization problem is formulated to maximize dynamic caching, forwarding, and wireless network resources considering limitations of resources.

Issues on network slicing optimization integrated with content networks are discussed in [34–36]. A Content Delivery Network as a Service (CDNaaS) platform is presented in [34] which focused on optimal placement of virtual machine and flavor selection for different images aiming at minimizing the cost of CDN slice owner and maximizing the quality of experience of streaming. In [35], the optimal content delivery strategy in cache-enabled HetNet is proposed by formulating a stochastic multicast scheduling problem to jointly minimize the average network delay and power costs taking into account the inherent multicast capability of wireless medium. In [36], a centrality-based caching algorithm is proposed in ICN based on complex network by exploiting the concept of (ego network) betweenness centrality to improve the caching gain and eliminate the uncertainty in the performance of the simplistic random caching strategy. A caching strategy based on betweenness centrality (*Betw* + *LRU*) and the approximation of it (*EgoBetw* + *LRU*) are also proposed for scalable and distributed realization in dynamic network environments where the full topology cannot be known a priori.

The influence of social network to caching strategy is addressed in [1, 37]. Since social relationship affects the success probability of a D2D communication, the performance analysis and radio resource allocation in social aware F-RAN contributes to the SE/EE (spectral efficiency/energy efficiency) and latency in F-RANs [1]. In [37], the heuristic algorithm PreCache for the selection of base stations is presented considering virtual spatial locality of content popularity for the change of content request probability by using a social influence model.

The research issues mentioned above contribute a lot to the network slicing on content-oriented services; however, content applications differentiate from communication services, which indicates that the factors affecting network slicing based on content caching in wireless access network not only lies in the resource optimization in RAN but also depends on some features concerning content caching including caching framework, caching policy, network scalability, and also performance metrics related to optimization objectives for caching resource allocation. From the viewpoint of content-centric resource allocation, although the cost of physical cache is becoming cheaper and cheaper, "Cache everything everywhere (CEE)" consumes a lot of redundant caching resources for copies of every contents in every cache in the network, and it also causes cache replacement error in the network [36]; therefore, the optimization of network slicing from the viewpoint of content is necessary. Since it aims at determining which contents should be cached in which nodes cooperatively, and content-oriented resources include cache resources and communication resources that used for transmitting contents, therefore, the network slice allocated to each content includes cache slice and communication slice. In order to provide good QoE for mobile users who require contents and improve network resource efficiency, how many content copies should be allocated and cached for a content slice in a scalable radio access network? And how about the communication resources should be allocated to the content for that content slice? To our best knowledge, it is still an open challenge.

In this paper, the network slicing and resource optimization on content-oriented application in cache-enabled hybrid radio access network based on complex network are investigated. The main contribution is summarized as follows:

(1) A cooperative caching framework named CNSC-RAN is proposed on content-oriented network slicing in hybrid radio access network, which is divided into MBS level, SBS level, and UE level. The functional modules including MBS controller, SBS controller, and UE controller are provided, and the procedures of network slicing including static slicing procedure and dynamic slicing procedure are proposed. The process to get the content required by UEs is designed in CNSC-RAN.

(2) Based on CNSC-RAN, content-oriented slicing is modeled and analyzed by using complex network which includes slicing on content cache resources and communication resources. In order to obtain the optimized resources sliced to each content, the optimization problem is formulated to minimize the system cost to get the content required by users in a known network architecture under ER model and BA model by an optimization algorithm called content-centric slicing optimization algorithm (CCSOA). The ideal weighted hop for one content is derived.

(3) In dynamic content-oriented network slicing procedure enabling UEs with self-evicting contents, the performance of CCSOA is evaluated by the metrics including hit rate, average cache occupation, average system cost, and request traffic reduction to MBS. The average copies and weighed communication cost for one content are got. Compared with CEE and ProbCache schemes, the simulation results indicate that the performance of CCSOA outperforms in hit rate, average system cost, and request traffic reduction when Zipf parameter less than 1 and achieves less average cache occupation when Zipf parameter is greater than 1.2.

The rest of this paper is organized as follows. In Section 2, the Cooperative Network Slicing Framework Based on Content in RAN (CNSC-RAN) is proposed. In Section 3, the system model is introduced. In Section 4, the content-oriented network slicing is formulated as an optimization problem for resource allocation aiming at minimizing the system cost for one content in the framework and solved by a heuristic algorithm called CCSOA. In Section 5, performance evaluation are provided and analyzed. Section 6 concludes the paper.

2 Cooperative network slicing framework based on content in RAN (CNSC-RAN)

In this section, a Cooperative Network Slicing Framework Based on Content in RAN (CNSC-RAN) is proposed for content-oriented slicing in cache-enabled hybrid radio access network.

2.1 System scenario

Considering the single macro cell scenario, in the system, a MBS, some SBSs, and UEs are deployed, and all of the equipment are cache-enabled. Each SBS connects to the MBS by fronthauls. Assuming communication links are deployed among SBSs, and D2D communication is supported for UEs, the system scenario of the cache-enabled hybrid radio access network is shown in Fig. 1.

The system scenario in Fig. 1 is divided into three levels, including MBS level, SBS level, and UE level. In MBS level, the MBS caches all the contents and provides content service in the macro cell. In the SBS level, the SBSs cache some of the contents and provide content service to the UEs in the small cells unless UEs cannot get the contents in the UE level. The SBSs in the SBS level cooperate with each other by links in order to get the contents required by UEs with minimum cost. In the UE level, each UE cache contents and provide contents for their neighbor UEs.

2.2 CNSC-RAN

In the system scenario shown in Fig. 1, a CNSC-RAN is proposed. In CNSC-RAN, the content-oriented slicing optimization controllers are resided in the SBSs and the MBS, which are cache enabled and responsible for content replacement optimization according to different optimization objectives. The network resources are allocated to each content according to the optimization result. Since content-oriented resources include cache resources and communication resources which are used for transmitting contents in the hybrid radio access network, the network slice allocated to each content includes cache slice and communication slice. The content-oriented slicing in the CNSC-RAN is shown in Fig. 2.

In CNSC-RAN, whenever a UE requests a content, firstly, the content request is sent to its neighbor UEs, if the request can be hit in the UE level, the content would be sent to the UE from one of its neighbor UEs which has the content in its cache required by the UE. Otherwise, the UE sends the content request to the SBS controller which covers the UE. If the SBS obtains the

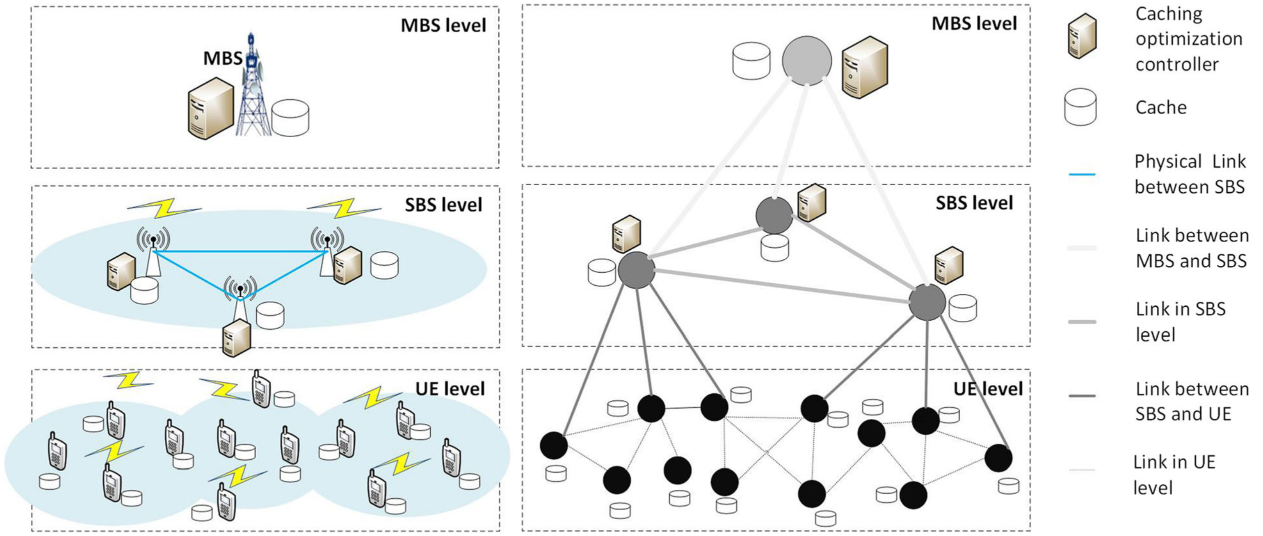

Fig. 1 System scenario of the cache-enabled hybrid radio access network

content either from its local cache or from its neighbor SBSs, then the SBS sends the content to the UE, and the request is hit in the SBS level. If the request cannot be satisfied in the SBS level, the request would be sent to the MBS by SBS. The MBS find the content in its local cache and sent it back to the UE, that is to say, the request can be hit finally by the MBS. When the request is hit in the SBS level, the content-oriented slicing optimization controller resided in the SBS which receives the request from the UE determines the optimal cache placement of the contents in the SBS level and UE level according to the content-centric slicing optimization algorithm (CCSOA). When the request is hit in the MBS, the content-oriented slicing optimization controller in the MBS determines the optimal cache placement of the contents in the SBS level and UE level by CCSOA.

In the following subsection, the content-oriented slicing in the CNSC-RAN is introduced in detail in the aspect of control plane, data plane, as well as functional modules of controllers and procedures of content-oriented network slicing.

2.2.1 Control plane

On the control plane, CNSC-RAN includes the Manager of Content-oriented Slicing Generation in RAN

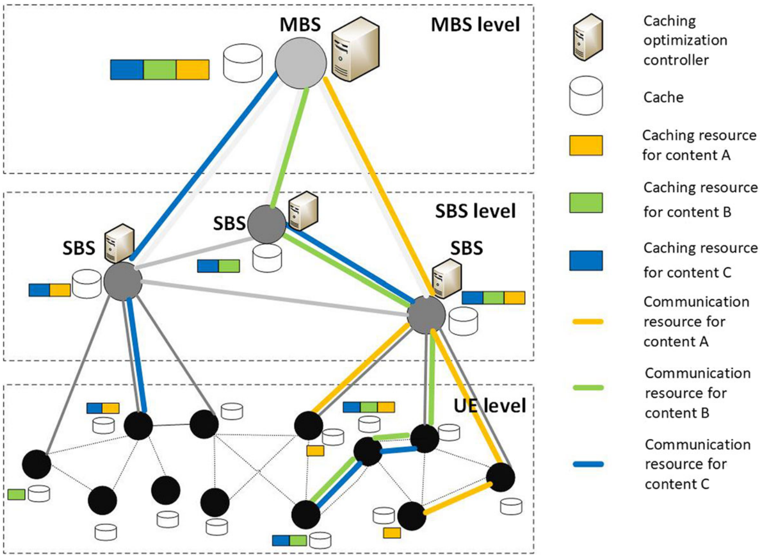
MBS
MBS level
SBS
SBS
SBS
SBS level
UE level
Caching optimization controller
Cache
Caching resource for content A
Caching resource for content B
Caching resource for content C
Communication resource for content A
Communication resource for content B
Communication resource for content C

Fig. 2 Content-oriented network slicing in CNSC-RAN

(MCSG), MBS controller, SBS controller, and UE controller. The controlling relationship of these controllers is shown in Fig. 3. MCSG is responsible for initialization, control management of CNSC-RAN, and network slicing information synchronization, while MBS controller, SBS controller, and UE controller are controllers in the MBS level, the SBS level, and the UE level respectively, which aim to control content-oriented slicing in CNSC-RAN.

2.2.2 Data plane

On the data plane, communication plane and cache plane are included in CNSC-RAN.

2.2.2.1 Communication plane On communication plane, the functional entities consist of the MBS, SBSs, UEs, two-way communication links between the MBS and SBSs, two-way communication links between SBSs, two-way communication links between SBSs and UEs, and two-way communication links between UEs. The two-way communication links between the MBS and SBS, as well as between two SBSs can either be wireless or wired.

2.2.2.2 Cache plane The functional entities in cache plane consist of the content caches deployed in the MBS, SBSs, and UEs. In CNSC-RAN, the content caches can be controlled by those controllers in upper levels for cache optimization replacement, and content caches deployed in UEs are also controlled by UEs themselves for content updating. In the caching schemes controlled by controllers in upper levels, the content caches in the UE level cache the contents determined by controllers in upper levels including SBS controllers and MBS controllers. In the caching schemes of self-controlled by UEs, the content caches in the UE level cache or evict contents by caching schemes such as LRU (least recently used) and LFU (least frequently used).

2.2.3 Functional modules of controllers in CNSC-RAN

Functional modules of MBS controller, SBS controller, and UE controller are illustrated in this section.

2.2.3.1 MBS controller The MBS controller controls the generation of content-oriented network slicing based on MBS level. The functional modules in MBS controller includes content service request receiver and response module, content-oriented network slicing management module, communication module, content-oriented network slicing optimization module on MBS level, and communication resource module. The functional modules in MBS controller which are used to generate content-oriented network slicing is shown in Fig. 4.

2.2.3.2 SBS controller The SBS controller controls the generation of content-oriented network slicing on SBS level. The functional modules in SBS controller includes content service request receiver and response module, content-oriented network slicing management module, communication module, content-oriented network slicing optimization module on SBS level, and communication resource module. The functional modules on generating content-oriented network slicing in SBS controller is shown in Fig. 5.

2.2.3.3 UE controller The UE controller takes charge of the generation of content-oriented network slicing on UE level. The functional modules in a UE controller includes message generation and receiving module, content obtainment module, content-oriented network slicing management module, communication module,

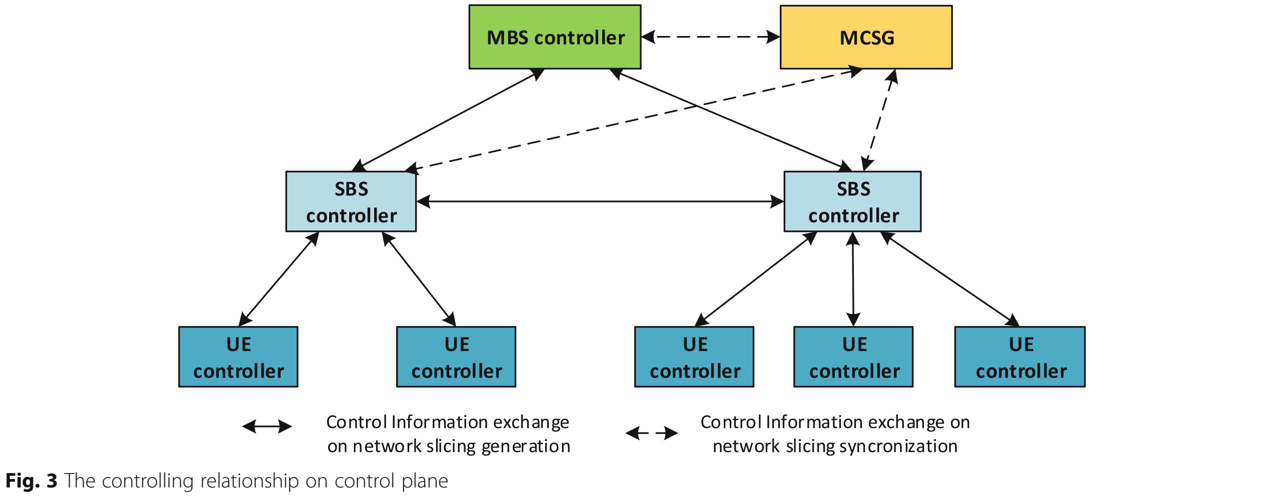
MBS controller
MCSG
SBS controller
SBS controller
UE controller
UE controller
UE controller
UE controller
UE controller
Control Information exchange on network slicing generation
Control Information exchange on network slicing syncronization

Fig. 3 The controlling relationship on control plane

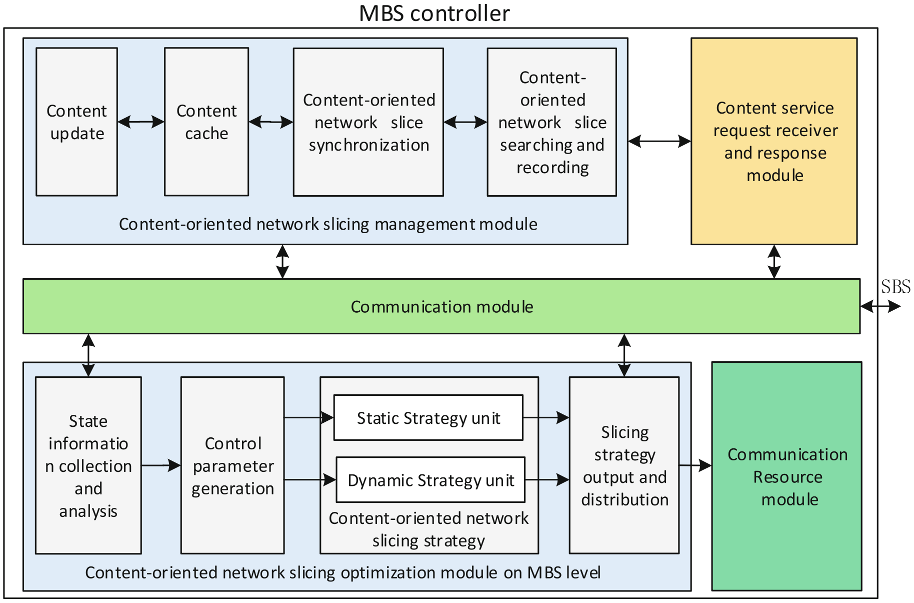
MBS controller
Content update
Content cache
Content-oriented network slice synchronization
Content-oriented network slice searching and recording
Content-oriented network slicing management module
Content service request receiver and response module
Communication module
SBS
State informatio n collection and analysis
Control parameter generation
Static Strategy unit
Dynamic Strategy unit
Content-oriented network slicing strategy
Slicing strategy output and distribution
Content-oriented network slicing optimization module on MBS level
Communication Resource module

Fig. 4 The functional modules in a MBS controller

state information collection and analysis module, and communication resource module. The functional modules in UE controller to generate the content-oriented network slicing is shown in Fig. 6.

2.2.4 Procedures of content-oriented network slicing in CNSC-RAN

In this section, the procedures of content-oriented network slicing are presented including request-driven dynamic network slicing and network performance-driven static network slicing.

2.2.4.1 Procedure of dynamic content-oriented network slicing Procedure of dynamic content-oriented network slicing is completed by either the sub-module of dynamic strategy unit in MBS controller or dynamic content-oriented network slicing strategy in SBS controller. When the MBS controller or one of the SBS controllers receives the content service request, the procedure of dynamic content-oriented network slicing can be started focusing on the optimization of the content caching required by the content service request. The optimization objectives and constraints are based on some specific metrics of network performance and/or

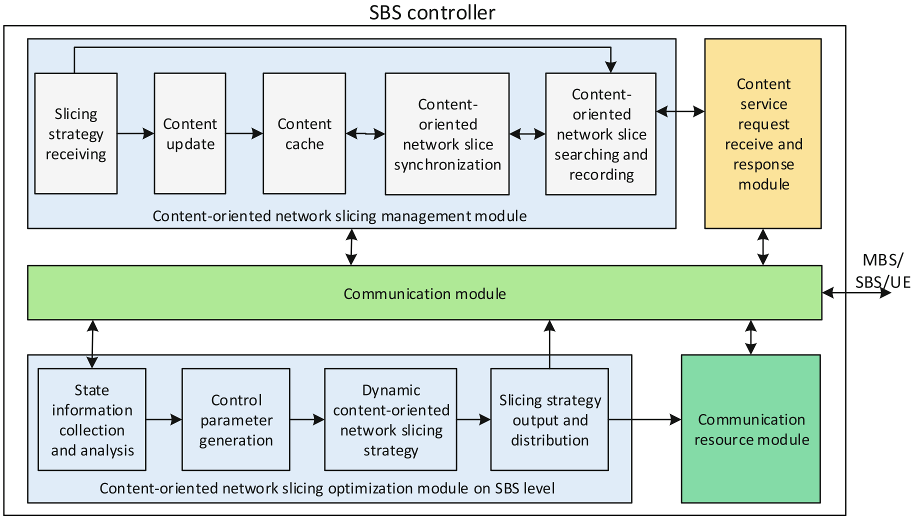
SBS controller
Slicing strategy receiving
Content update
Content cache
Content-oriented network slice synchronization
Content-oriented network slice searching and recording
Content-oriented network slicing management module
Content service request receive and response module
Communication module
MBS/ SBS/UE
State information collection and analysis
Control parameter generation
Dynamic content-oriented network slicing strategy
Slicing strategy output and distribution
Content-oriented network slicing optimization module on SBS level
Communication resource module

Fig. 5 The functional modules in a SBS controller

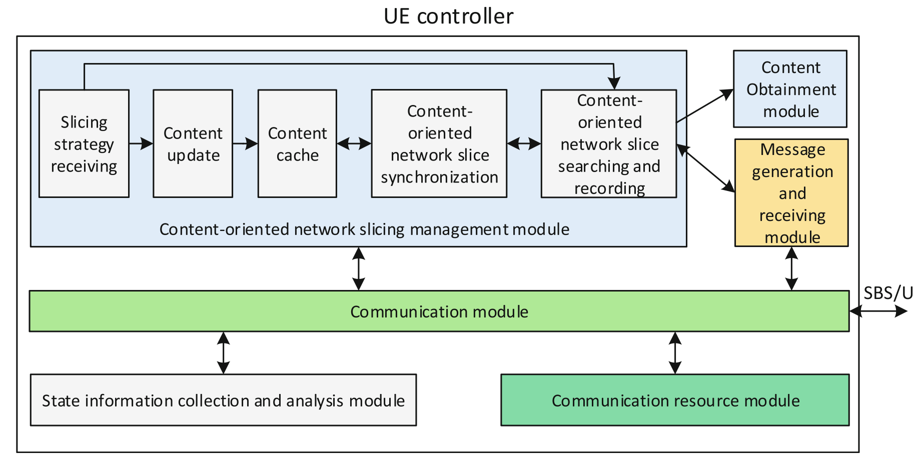
UE controller
Slicing strategy receiving
Content update
Content cache
Content-oriented network slice synchronization
Content-oriented network slice searching and recording
Content Obtainment module
Message generation and receiving module
Content-oriented network slicing management module
Communication module
SBS/UE
State information collection and analysis module
Communication resource module

Fig. 6 The functional modules in a UE controller

quality of content service. The procedure of dynamic content-oriented network slicing works as follows:

Step 1: The content service request sent by the UE is received, and the content identification information is abstracted, then go to step 2.

Step 2: State information collection and analysis sub--module analyzes the state information and output the control parameters and network parameters for the network slicing optimization to the control parameter generation sub-module. The control parameter generation sub-module generates the input parameters, then go to step 3.

Step 3: Based on the input parameters, the sub-module of dynamic strategy unit/dynamic content-oriented network slicing strategy formulates and resolves the optimization problem on optimal replacement of the required content by selecting a specific metric as the optimization object. The optimal content-oriented resource allocation results including caching resources and communication resources are obtained, which is the result of content-oriented network slicing for the content service request. The network slicing result information for the content is output to the slicing strategy output and distribution sub-module, then go to step 4.

Step 4: Slicing strategy output and distribution sub--module distributes the network slicing result information for the required content to the related SBS controllers and UE controllers, then go to step 5.

Step 5: The content-oriented network slicing management modules in SBSs and UEs slice the cache resources, and the communication resource modules in SBSs and UEs slice the communication resources according to the network slicing result information, then go to step 6.

Step 6: The content-oriented network slice for the required content is generated, and the procedure of dynamic content-oriented network slicing ends.

2.2.4.2 Procedure of static content-oriented network slicing

The procedure of static content-oriented network slicing is completed by the static strategy unit in the MBS controller. The MBS controller monitors the state information of specific metrics based on network performance and/or quality of service in CNSC-RAN periodically. If the state information of specific metrics on network performance cannot meet the requirements of content service provider, the MBS controller starts the static content-oriented network slicing based on all of the contents or part of the contents in CNSC-RAN.

The detailed procedure of static content-oriented network slicing is as follows:

Step 1: The state information collection and analysis sub-module collects and analyzes the state information in CNSC-RAN and computes the selected metrics of network performance and/or quality of content service periodically.

Step 2: The state information collection and analysis sub-module make a decision on whether the selected metric value meets the requirements of network performance and/or quality of content service, if the selected metric value meets the requirements of network performance and/or the quality of content service, then go to step 1; otherwise, go to step 3.

Step 3: Control parameter generation sub-module generates the optimization parameters according to the input of the state information collection and analysis sub-module, then go to step 4.

Step 4: Static strategy unit sub-module formulates the optimization problem as the optimal caching replacement of all the contents/part of the contents in CNSC-RAN by selecting the specific metric as the optimization object, and it obtains the optimization result of content-oriented network slicing including caching resources and communication resources. The optimization results are output as the content-oriented network slicing result information for each content to satisfy the optimal value of the selected metric of network performance and/or quality of content service, then go to step 5.

Step 5: Slicing strategy output and distribution submodule distributes the content-oriented network slicing result information for each content to related SBSs and UEs, then go to step 6.

Step 6: According to the content-oriented network slicing result information, the content-oriented network slicing management modules in SBSs and UEs slice the cache resources, and the communication resource modules in SBSs and UEs slice the communication resources, then go to step 7.

Step 7: The content-oriented network slicing for all the contents/part of the contents are generated, and the procedure of static content-oriented network slicing ends.

2.2.4.3 Content-oriented network slice synchronization

In CNSC-RAN, the caching resources allocated to some contents often changes due to UE mobility and content cache eviction by UEs, which leads to the obsolete content-oriented network slicing information in content-oriented network slicing management modules in SBSs and UEs. Therefore, content-oriented network slice synchronization is necessary.

Whenever the content cached in UE is evicted or UE moves out of the covered area of CNSC-RAN, the UE controller transmits the content-oriented network slice synchronization information about these events to the SBS controller it connects in advance. The SBS controller integrates the information about these events from UEs and sends the request of content-oriented network slice synchronization for some contents to MCSG. MCSG extracts the content-oriented network slice synchronization information which are need to be synchronized and distributes them as the requests of content-oriented network slice synchronization updating information to all the SBSs, UEs, and MBS. The MBS controller, SBS controllers, and UE controllers update the content-oriented network slice information in their content-oriented network slice searching and recording sub-modules.

3 Modeling of CNSC-RAN

In this section, the system model of CNSC-RAN is presented, including content request model, network model based on complex network, and content search model.

3.1 Content request model

In content-oriented cache-enabled networks, content request model is usually modeled by Zipf law [23]. Assume that all the contents are the same size and each UE has the same interest, according to Zipf law, the probability that the content ranked f is requested by a UE is given by $\frac{P_f=f^{-\alpha}}{\sum_{i=1}^{F} i^{-\alpha}}$, where F is the number of contents and $\alpha \geq 0$ is the parameter based on statistics that describes the skewness of content popularity. When $\alpha = 0$, all the contents have the same popularity. The higher α is, the more probable the popular contents are to be requested.

With the enormous development of the self-media service on mobile internet, the contents published by users become more and more, and this would lead to the variety of the contents in the network, which means that the Zipf parameter α tends to be small.

3.2 Network model based on complex network

The system model in Fig. 1 can be abstracted as an undirected graph $G(V,E)$, in which V is denoted as the set of nodes which are UEs, SBSs, and MBSs, and E is denoted as the set of edges in the graph which are described as the communication links among network equipment. According to Fig. 1, the graph G can also be divided into three levels including UE level, SBS level, and MBS level. In the UE level, the number of nodes is N_{UE}. In the SBS level, the number of nodes is N_{SBS}. In order to differentiate the resource importance of the links among different nodes, the weight of the communication links are set, namely the weight of the edge is w_1 in the UE level, the weight of the edge is w_2 between UE level and SBS level, and the weight of the edge is w_3 in the SBS level. The weight of the edge is w_4 between SBS level and MBS level.

In the UE level, d_i^{UE} is denoted as the degree value of node i, so the average degree value of nodes in the UE level can be written as:

$$\overline{d_{\mathrm{UE}}} = \frac{\sum_{i=1}^{N_{\mathrm{UE}}} d_i^{\mathrm{UE}}}{N_{\mathrm{UE}}} \tag{1}$$

Since UEs move randomly in the area and communicate with other UE through D2D, the link between node i and node j in the UE level can be seen as a random link which is expressed as h_{ij}^{UE}, then,

$$h_{ij}^{\mathrm{UE}} = \begin{cases} 1, & \text{If node i connects with node } j \\ 0, & \text{Otherwise} \end{cases}$$

Let the matrix H^{UE} indicate the connection of UE level.

Between UE level and SBS level, let s_i indicate the ID of SBS connected to UE i, and b_i indicate the number of UEs connected to SBS i. $s_i = k$ indicates that UE i connects to SBS k and we denote

$$H_{ji}^{SBS-UE} = \begin{cases} 1, & \text{if } j = k \\ 0, & \text{if } j \neq k \end{cases}$$

In the SBS level, we use the matrix H^{SBS} to indicate the connection of SBS level. The connection between node i and node j is denoted as h_{ij}^{SBS}.

$$h_{ij}^{\mathrm{SBS}} = \begin{cases} 1, & \text{If node } i \text{ connects with node } j \\ 0, & \text{Otherwise} \end{cases}$$

In order to investigate the network slicing of each content in a scalable heterogeneous radio access network, random network model based on complex network can be used to describe the practical network among nodes. Many models are presented to illustrate the features of the random network architectures in reality in complex network theory. Among them, two models are most widely used to emulate connection behaviors of the mobile users based on the average degree value in radio access networks, which are BA (Barabasi-Albert) model and ER (Erdos-Renyi) model [38]. In BA model, only a few nodes have very high degree value compared with other nodes, while the vast majority of nodes have low degree value; therefore, those nodes with high degree value are pretty well-known nodes which many nodes would prefer to connect to. In ER model, every node nearly has the same degree value, which is similar to the scenario of random connection among users in the RAN.

3.3 Content search model

In content search model, upon the content *c* is requested by UE S1, different decision process would be made depending on where the content is cached. First of all, the content c is searched in the UE level. If the request cannot be met in the UE level, the content c would be searched in the SBS level. If the request cannot be met in the SBS level as well, the request for content *c* is then sent to MBS in MBS level. Let the cost to get the content *c* is *w*, which indicates the sum of weighted hops to get the content, then the detailed search process is described for getting the content required by UEs in CNSC-RAN as follows.

(1) Step 1: The UE S1 searches its own cache. If the request is hit, then the cost to get the content *c* in this step is zero, otherwise, enter step 2.

(2) Step 2: The UE S1 searches the content in its ego network where the UE S1 has the caching information of the neighbors. If the request is hit, then the process is over, and the cost to get the content *c* in this step is the edge weight w_1; otherwise, add w_1 for the cost and enter step 3.

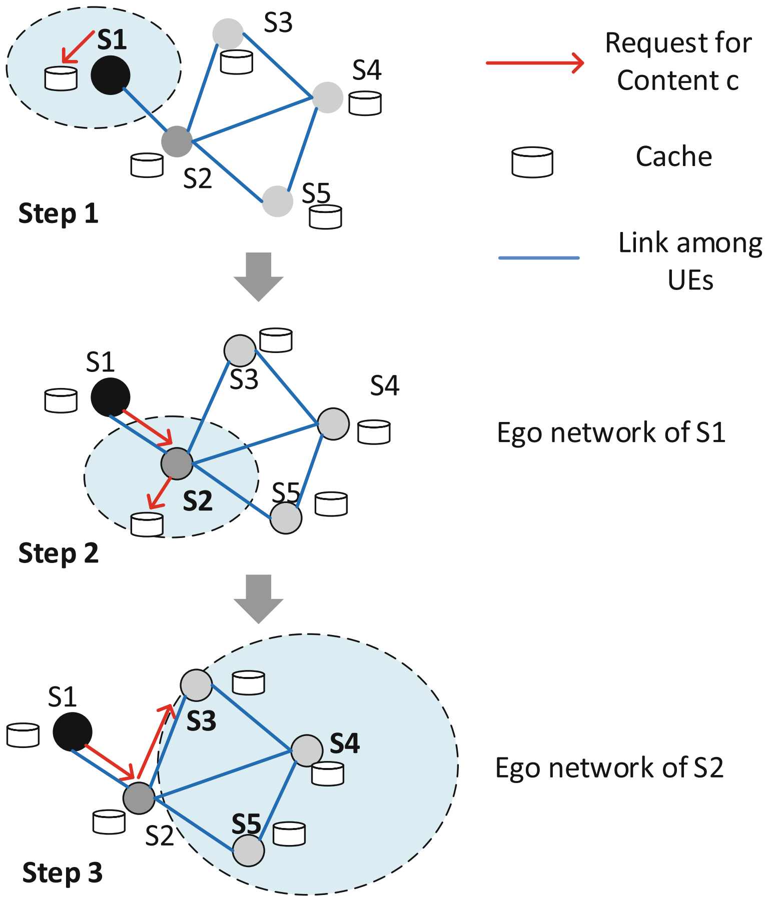

Fig. 7 A typical process of content search model of UE S1 in the UE level

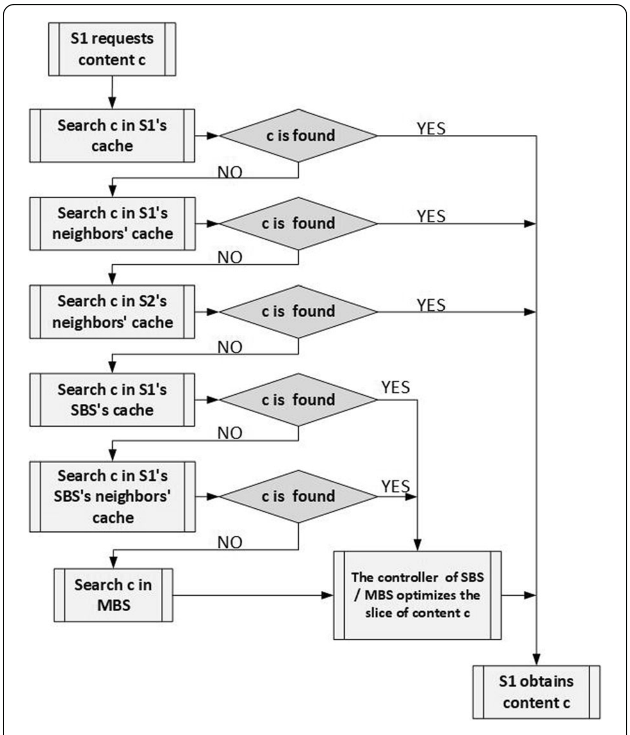
S1 requests content c
Search c in S1's cache
c is found
YES
NO
Search c in S1's neighbors' cache
c is found
YES
NO
Search c in S2's neighbors' cache
c is found
YES
NO
Search c in S1's SBS's cache
c is found
YES
NO
Search c in S1's SBS's neighbors' cache
c is found
YES
NO
Search c in MBS
The controller of SBS / MBS optimizes the slice of content c
S1 obtains content c

Fig. 8 The process of request and cache decision for content c

(3) Step 3: The UE S1 sends its request to neighbor UE S2 who has the greatest degree value in the ego network to search the content c in S2's ego network (in Fig. 3, including UE S3, S4, and S5). If the request is hit, then the cost in this step is the edge weight w_1, and the process is over; otherwise, add w_1 for the cost and enter step 4.

(4) Step 4: When the request is not hit in the UE level, the UE S1 sends the content request to its connected SBS s_{S1}, if the request is hit in the local cache in the SBS, the process ends and the cost in this step is w_2; otherwise, add w_2 for the cost and enter step 5;

(5) Step 5: The SBS s_{S1} searches for the content c in its neighbor SBSs in the SBS level, if the content is hit, then the process is over, and the cost in this step is w_3; otherwise, add w_3 for the cost and enter step 6;

(6) Step 6: The SBS s_{S1} requests the MBS for the content c, which has all of the contents provided to the UEs in its coverage area, and the cost in this step is w_4.

A typical process of content search model of UE S1 in the UE level is shown in Fig. 7 including step 1, step 2, and step 3. The flowchart of the decision process for the content c by UE S1 is depicted in Fig. 8.

Based on the process described above, the cost to obtain the content required by the UE is modeled by the sum of weighted hop from UEs, SBSs, and MBS. The total cost to hit the content request can be got as (2),

$$R_i = \begin{cases} 0\ , & i = 1 \\ w_1, & i = 2 \\ 2w_1, & i = 3 \\ 2w_1 + w_2, & i = 4 \\ 2w_1 + w_2 + w_3, & i = 5 \\ 2w_1 + w_2 + w_3 + w_4, & i = 6 \end{cases} \tag{2}$$

Where $R_i(i = 1, 2, 3, 4, 5, 6)$ is the total cost to get the content when the request is hit in step i. In order to get the content required by UE at the minimum cost, it can be formulated as an optimization problem which aims to minimize the sum of the weighted hop to get the content.

4 Problem formulation

In this section, a content-centric slicing optimization algorithm (CCSOA) is proposed to find the optimal cache places so as to slice the cache resources and communication resources for each content in CNSC-RAN.

4.1 System cost for one content

In CNSC-RAN, the cache optimization controllers on the MBS determine the content caching placement in the UE level and SBS level. Assuming the case which there is only one content in the framework, the sum of the cost of all UEs to obtain the content is selected as the optimization objective (which is called the system cost later). Obviously, it should be minimized.

Assuming that all the UEs request the same content c, then the system cost for the content c can be obtained as,

$$E^c = \sum\nolimits_{i=1}^{6} R_i N_i \tag{3}$$

Where N_i is the number of the UEs whose requests are hit in ith step. Let C_i^{SBS} be a binary variable, that is to say,

$$C_i^{\mathrm{SBS}} = \begin{cases} 1, & \text{If the content is cached in SBS } i \\ 0, & \text{otherwise} \end{cases}$$

We denote C_i^{UE} as a binary variable, and then

$$C_i^{\mathrm{UE}} = \begin{cases} 1, & \text{If one content is cached in UE } i \\ 0, & \text{otherwise} \end{cases}$$

In addition, we denote d_i as the identifier of the node who has the greatest degree value among the neighbors of UE i in the UE level, so the cost of the system in each step can be modeled according to the content search model.

In step 1, the UE who requires the content c searches the content c in its local cache, and if the request is hit, then the number N_1 of UEs whose request is hit in this step is expressed as:

$$N_1 = \sum\nolimits_{i=1}^{N_{\mathrm{UE}}} C_i^{\mathrm{UE}} \tag{4}$$

In step 2, if the request cannot hit in step 1, then the UE searches the content around its neighbors, and if the request is hit, then the number N_2 of UEs whose request is hit in this step can be expressed as:

$$N_2 = \sum\nolimits_{i=1}^{N_{\mathrm{UE}}} (1-C_i^{\mathrm{UE}})(\sum\nolimits_{\mathrm{Binary},j=1}^{N_{\mathrm{UE}}} C_j^{\mathrm{UE}} H_{ij}^{\mathrm{UE}}) \tag{5}$$

In step 3, if the request cannot be hit in step 2, the neighbor UE who has the greatest degree value in step 2 would search the content in its ego network. The number N_3 of UEs whose request is hit in this step is expressed as:

$$\begin{aligned} N_3 = &\sum\nolimits_{i=1}^{N_{\mathrm{UE}}} (1-C_i^{\mathrm{UE}})(\prod\nolimits_{j=1}^{N_{\mathrm{UE}}} (1-C_j^{\mathrm{UE}}) H_{ij}^{\mathrm{UE}}) \\ &\times (\sum\nolimits_{\mathrm{Binary},k=1}^{N_{\mathrm{UE}}} C_k^{\mathrm{UE}} H_{\boldsymbol{d}_i} k^{\mathrm{UE}}) \end{aligned} \tag{6}$$

In step 4, if the request cannot be hit yet, then the UE requests the SBS it connected with for the content. The number N_4 of UEs whose request is hit in this step is expressed as:

$$\begin{aligned} N_4 = &\sum\nolimits_{i=1}^{N_{\mathrm{UE}}} (1-C_i^{\mathrm{UE}}) \\ &\times (\prod\nolimits_{j=1}^{N_{\mathrm{UE}}} (1-C_j^{\mathrm{UE}}) H_{ij}^{\mathrm{UE}}) C_{\boldsymbol{s}_i}^{\mathrm{SBS}} (\prod\nolimits_{k=1}^{N_{\mathrm{UE}}} (1-C_k^{\mathrm{UE}}) H_{\boldsymbol{d}_i} k^{\mathrm{UE}}) \end{aligned} \tag{7}$$

In step 5, if the request cannot be hit in the local cache of the SBS, the SBS searches the content in the caches of its neighbor SBSs. The number N_5 of UEs whose request is hit in this step is expressed as:

$$\begin{aligned} N_5 = &\sum\nolimits_{i=1}^{N_{\mathrm{UE}}} (1-C_i^{\mathrm{UE}})(\prod\nolimits_{j=1}^{N_{\mathrm{UE}}} (1-C_j^{\mathrm{UE}}) H_{ij}^{\mathrm{UE}}) \\ &\times (1-C_{\boldsymbol{s}_i}^{\mathrm{SBS}})(\prod\nolimits_{k=1}^{N_{\mathrm{UE}}} (1-C_k^{\mathrm{UE}}) H_{\boldsymbol{d}_i} k^{\mathrm{UE}}) \\ &\times (\sum\nolimits_{\mathrm{Binary},q=1}^{N_{\mathrm{SBS}}} C_q^{\mathrm{SBS}} H_{\boldsymbol{s}_i} q^{\mathrm{SBS}}) \end{aligned} \tag{8}$$

In step 6, if the request cannot be hit in the caches of the neighbor SBSs, the SBS sends the UE request to the MBS for the content. The number N_6 of UEs whose request is hit in this step is expressed as:

$$\begin{aligned} N_6 = &\sum\nolimits_{i=1}^{N_{\mathrm{UE}}} (1-C_i^{\mathrm{UE}})(\prod\nolimits_{j=1}^{N_{\mathrm{UE}}} (1-C_j^{\mathrm{UE}}) H_{ij}^{\mathrm{UE}}) \\ &\times (1-C_{\boldsymbol{s}_i}^{\mathrm{SBS}})(\prod\nolimits_{k=1}^{N_{\mathrm{UE}}} (1-C_k^{\mathrm{UE}}) H_{\boldsymbol{d}_i} k^{\mathrm{UE}}) \\ &\times (\prod\nolimits_{q=1}^{N_{\mathrm{SBS}}} (1-C_q^{\mathrm{SBS}}) H_{\boldsymbol{s}_i} q^{\mathrm{SBS}}) \end{aligned} \tag{9}$$

The optimization of the system cost can be formulated as:

$$\mathrm{Min} E^c \tag{10}$$

$$\text{s.t.} \ \sum\nolimits_{i=1}^{N_{\mathrm{UE}}} C_i^{\mathrm{UE}} \le \Gamma \tag{11}$$

$$\sum\nolimits_{i=1}^{N_{\mathrm{UE}}} \sum\nolimits_{j=1}^{N_{\mathrm{UE}}} C_i^{\mathrm{UE}} C_j^{\mathrm{UE}} H_{ij}^{\mathrm{UE}} = 0 \tag{12}$$

$$\sum\nolimits_{i=1}^{N_{\mathrm{SBS}}} \sum\nolimits_{j=1}^{N_{\mathrm{SBS}}} C_i^{\mathrm{SBS}} C_j^{\mathrm{SBS}} H_{ij}^{\mathrm{SBS}} = 0 \tag{13}$$

$$C_i^{\mathrm{UE}} \in \{0,1\}, i \in [1, N_{\mathrm{UE}}] \tag{14}$$

$$C_i^{\mathrm{SBS}} \in \{0,1\}, i \in [1, N_{\mathrm{SBS}}] \tag{15}$$

In order to reduce the cache redundancy for the content c, constraints (12) and (13) are used to restrict that one content is cached both in neighbor UEs and neighbor SBSs, and the parameter Γ in (11) is the maximum copies of one content in the UE level.

From the view of complex network theory, since some nodes have great centrality, then the contents are preferred to be cached in those nodes with great centrality so that the hit rate for the contents is high. The factors affecting the hit rate of the system include the average degree value of the graph, the connectivity feature of the graph (ER model or BA model), the scale of the network (the number of nodes), the number of copies of the contents caching in the network, and the weight value for each hop in/between different level as well as the popularity of the contents and caching policy of each node in the network. With the formulation (10–15), if the hit rate in the UE level is greater, then the system cost E^c is smaller, in other words, it is better that the contents can be hit in the UE level. However, contents cannot be cached in every UE because of the limited caching space in UE, the challenge is to find the tradeoff between the hit rate and the system cost. With the appropriate value of Γ, the optimal caching placement with minimal system cost and high hit rate in the UE level can be found.

4.2 Caching insertion and eviction policy for a content

In order to investigate the ideal number of maximum copies of each content in the system and avoid the influence caused by content eviction often used by caching policies for content-oriented cache management, we assume that the cache size in each UE and SBS is not limited, that is to say, the caching insertion policy is that each node has enough cache space to store various kinds of contents, and the cache eviction policy is removing the contents periodically.

Assuming that the cache eviction period of UEs is T_{UE} and cache eviction period of SBS is T_{SBS}, which means that the content can be cached in the UEs during T_{UE} and T_{SBS} in SBSs, respectively. The cache time of content i is t_i. Once the content in cache is requested, t_i will be reset to zero. If $t_i > T_{\mathrm{UE}}$ in the UE level, the content in UE cache will be evicted. If $t_i > T_{\mathrm{SBS}}$ in the SBS level, the content in SBS cache will be evicted.

According to the abovementioned caching insertion and eviction policy, by solving the optimization problem, the minimal system cost to obtain one content in the system can be got to match the highest hit rate in the UE level, so the resources allocated to one content can be optimized, which is the network slice allocated to one content including caching resource and communication resource for the content.

4.3 The algorithm of CCSOA

CCSOA is the algorithm used by SBS controller to slice the resources to contents, including caching placement for contents, whose optimization objective is the total system cost to get the contents required by UEs in CNSC-RAN.

Based on the content request model, content search model and caching policy in CNSC-RAN, the algorithm of CCSOA can be described as following:

```
Algorithm 1: CCSOA for content c
Input parameters: $target\_hit\_rate, H^{SBS}, H^{UE}, H^{SBS-UE}, w_1, w_2, w_3, w_4$
$\Gamma \leftarrow 0, optimized\_hit\_rate \leftarrow 0,$
Do
  $\Gamma \leftarrow \Gamma + 1$
  //Genetic algorithm:
  while $\mathrm{pop.size} \le \mathrm{M}$
    Generate $C^{UE} \leftarrow \{C_1^{UE}, C_2^{UE}, \dots\dots, C_{N_{UE}}^{UE}\}, C^{SBS} \leftarrow \{C_1^{SBS}, C_2^{SBS}, \dots\dots, C_{N_{SBS}}^{SBS}\}$
    Satisfying $\sum_{i=1}^{N_{UE}} \sum_{j=1}^{N_{UE}} C_i^{UE} C_j^{UE} H_{ij}^{UE} = 0$ and $\sum_{i=1}^{N_{SBS}} \sum_{j=1}^{N_{SBS}} C_i^{SBS} C_j^{SBS} H_{ij}^{SBS} = 0$ and
    $\sum_{i=1}^{N_{UE}} C_i^{UE} \le \Gamma$
    insert $\{C^{UE}, C^{SBS}\}$ into pop
    compute $E^c$
    insert $E^c$ into Val
  end while
  loop$\leftarrow$ 2
  while loop $\le$ G
    while $\mathrm{newpop.size} \le \mathrm{M}$
      select two elements in pop according to the proportion of Val
      if random(0,1)< $P_c$
        cross the two elements and update the two elements
      if random(0,1)< $P_m$
        variation of the two elements and update the two elements
      evict the element which does not satisfy
        $\sum_{i=1}^{N_{UE}} \sum_{j=1}^{N_{UE}} C_i^{UE} C_j^{UE} H_{ij}^{UE} = 0$ and $\sum_{i=1}^{N_{SBS}} \sum_{j=1}^{N_{SBS}} C_i^{SBS} C_j^{SBS} H_{ij}^{SBS} = 0$ and
        $\sum_{i=1}^{N_{UE}} C_i^{UE} \le \Gamma$
      Insert the remaining elements into newpop
      Compute $E^c$ of the remaining elements and insert into newVal
    end while
    pop $\leftarrow$ newpop, Val $\leftarrow$ newVal
    clear newpop
    loop $\leftarrow$ loop + 1
  end while
  //genetic algorithm end
  find minimization in Val
  Get $C^{UE}, C^{SBS}$
  Compute $\{N_1, N_2, N_3, N_4, N_5, N_6\}$
  Compute $Path\_set \leftarrow \{v_1, v_2, \dots, v_{N_{UE}}\}$
  $optimized\_hit\_rate = \sum_{i=1}^{3} N_i / \sum_{i=1}^{6} N_i$
Until $optimized\_hit\_rate \ge target\_hit\_rate$
Output: $\Gamma, C^{UE}, C^{SBS}, E^c, optimized\_hit\_rate, Path\_set$
```

where P_c is the crossing probability in genetic algorithm, P_m is the probability of occurrence of variation, M is the population size, and G is the evolutionary generations for genetic algorithm. Path_set is the set of v_i which denotes as the path of node i to find the content required by node i.

5 Performance evaluation

In this section, CCSOA is solved with genetic algorithm to investigate the impact of the parameters including Γ, α, $\overline{d_{\mathrm{UE}}}$, number of the nodes, and connectivity features of network (ER model and BA model) to performance metrics of CNSC-RAN.

The performance of the proposed algorithm CCSOA is evaluated by MATLAB with Monte Carlo method. The network in the UE level is generated either by ER model or BA model. The content request arrival process follows a Poisson process, with the average request rate per UE γ. According to CNSC-RAN, the content request is searched in the UE level first. If it cannot be hit in the UE level, the request will be sent to the SBS level. If it cannot be hit in the SBS level, the request will be sent to the MBS level. When the request is sent to the MBS level, the MBS controller decides the optimal content placement by solving the optimization problem using CCSOA and satisfies the content request from caches in the SBS level or the cache in MBS. The performance metrics of the system include hit rate, average cache occupation, average system cost, and traffic load reduction, which are defined in Section 5.2 respectively.

The baseline caching insertion policies compared with CCSOA include CEE (Cache Everything Everywhere) and ProbCache (Probability Cache). The caching eviction policy in both baselines is LRU (Least Recently Used). The probability of content cached in ProbCache is denoted by ρ.

5.1 Network model and simulation parameters used in the evaluation

In this section, the complex networks in the UE level are generated including ER model and BA model, and the frequency histogram of degree value in the UE level is shown. The parameters used in the simulations are given.

5.1.1 Network model based on complex network

The connection relation of UEs in the UE level is modeled by ER model and BA model. In the SBS level, each SBS has connections with each other. In order to generate the connection relation between UEs and SBSs, the nodes in the UE level are clustered, and the clusters are associated to SBSs in the SBS level randomly. One of the degree value frequency histograms for ER model and BA model

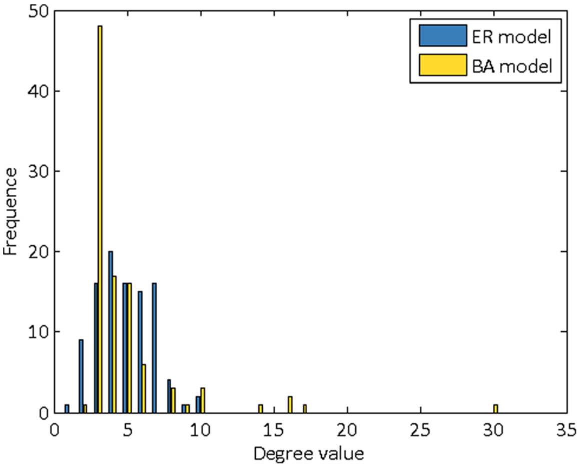

Fig. 9 Frequency histogram of degree value ($\overline{d_{\mathrm{UE}}} = 4, N_{\mathrm{UE}} = 100$)

generated for simulation is shown in Fig. 9. In BA model, there are a few nodes connecting with over ten nodes, while the vast majority of nodes connecting with about 3 nodes. In ER model, every node nearly connects with 2 to 7 nodes, which is similar to the scenario of random connection among users in the RAN.

5.1.2 Simulation parameters used in the performance evaluation

In the simulation, assuming that the size of all the contents are the same, so the cache occupied by the contents can be denoted as the copies of one content. There are $F = 1000$ contents in the system which are provided to UEs, and they can be cached in the UE level and the SBS level. The time for simulation is 20 min, the average request rate per UE is $\gamma = 6$ requests/min. The parameters used in the simulations are listed in Table 1.

Table 1 Simulation parameters

Parameter	Value in the simulation
N_{UE}	20–100
N_{SBS}	3
F	1000
T_{SBS}	0 – 10 min
T_{UE}	1min
w_1	1
w_2	3
w_3	1
w_4	5
ρ	0.75
α	0–2
Γ	0–10

5.2 Performance metrics used in the performance evaluation

The performance of CNSC-RAN exploiting CCSOA is evaluated, focusing on the influences of the following six factors to the algorithm: (1) Γ, the maximum number of copies of one content in the caches in the UE level; (2) content popularity, namely Zipf parameter α; (3)network type, BA model, and ER model are selected, which indicate the connection features of UE network in the UE level;(4) the average degree value of network in the UE level; (5) N_{UE}, the number of UEs; and (6) ${}^{T_{\mathrm{SBS}}}\!/_{T_{\mathrm{UE}}}$, which indicates the impact of caching eviction policy to the performance of the system in different levels.

Four performance metrics are used to evaluate the performance of CNSC-RAN with CCSOA, including hit rate, average cache occupation, average system cost, and request traffic reduction to MBS. The hit rate and average cache occupation are used to evaluate the performance of caching resources allocated to the contents and also the QoE to provide content-oriented service by operators. The average system cost aims to evaluate the communication resources allocated to the contents, and request traffic reduction to MBS aims to measure the request load reduction to MBS and communication load reduction from MBS to SBSs for content downloading in the system. The definition of the four metrics is illustrated bellow.

(1) Hit rate(HT)

Hit rate usually refers to the ratio of the content requests met by the system among total content requests required by users in the system. In CNSC-RAN, it includes the hit rate in the UE level and the hit rate in the SBS level, which are defined in (16) and (17) as,

$$\mathrm{HT_{UE}} = \frac{\mathrm{hit\_num\_UE}}{\mathrm{total\_req\_num}} \tag{16}$$

$$\mathrm{HT_{SBS}} = \frac{\mathrm{hit\_num\_SBS}}{\mathrm{req\_num\_to\_SBS}} \tag{17}$$

where total_req_num denotes as the total number of the requests from the UE level, hit_num_UE indicates the number of the requests which are hit in the UE level, hit_num_SBS indicates the number of the requests which are hit in the SBS level and req_num_to_SBS indicates the number of requests which are sent to the SBS level.

(2) Average Cache Occupation(ACO)

The metric of average cache occupation is used to measure the average cache occupation of contents for every

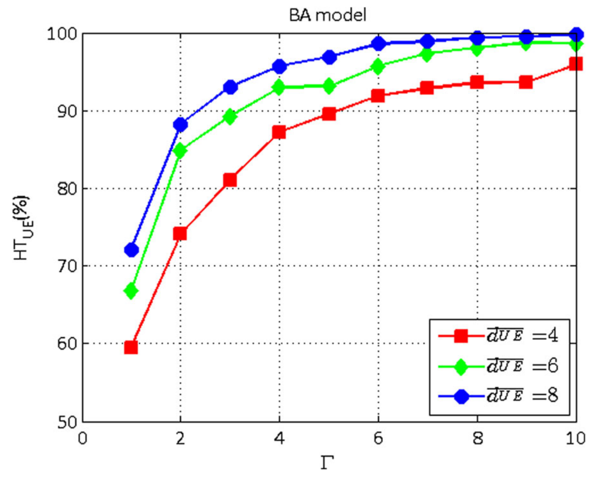

Fig. 10 Impact of average degree value to hit rate in BA model ($N_{UE} = 100$)

node in a network. It includes average cache occupation in the UE level, and average cache occupation in the SBS level which are defined in (18) and (19) as follows:

$$\mathrm{ACO_{UE}} = \frac{\mathrm{total\_cache\_occu\_UE}}{\mathrm{N_{UE}}} \tag{18}$$

$$\mathrm{ACO_{SBS}} = \frac{\mathrm{total\_cache\_occu\_SBS}}{\mathrm{N_{SBS}}} \tag{19}$$

Where total_cache_occu_UE is the total content cache occupation in the UE level, and total_cache_occu_SBS indicates the total content cache occupation in the SBS level.

(3)Average System Cost(ASC)

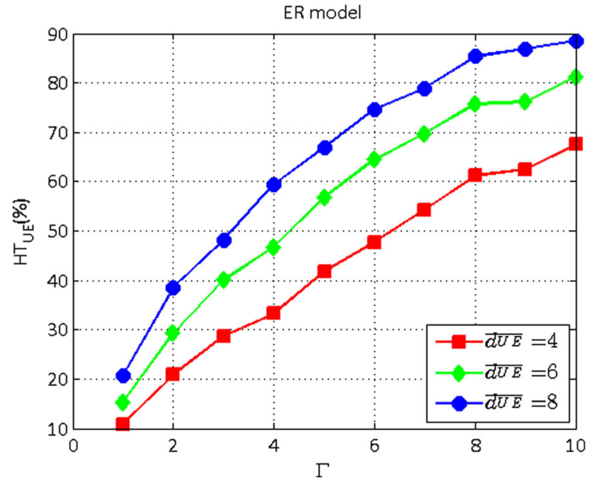

Fig. 11 Impact of average degree value to hit rate in ER model ($N_{UE} = 100$)

The average system cost is defined to measure the average weighted hops to get the contents in the system for every request, namely as follows in (20):

$$\mathrm{ASC} = \frac{\mathrm{total\_cost}}{\mathrm{total\_req\_num}} \tag{20}$$

where total_cost indicates the total cost to meet all the requests in the system, which is the total weighted hops to get the contents in the system. Obviously,the average system cost can be regarded as the communication resources allocated for transmitting one content for every content request.

(4)Request Traffic Reduction to MBS(RTR)

Request Traffic Reduction to MBS is defined as a metric in control plane to evaluate the request traffic reduction to MBS in the system. More requests are hit in the UE level or the SBS level would alleviate the content request traffic load to MBS, and at the same time lead to reduction of communication traffic load sliced for content downloading from MBS to SBS, because all the contents can be found in MBS. RTR is defined as follows in (21):

$$RTR = \frac{hit\_num\_UE + hit\_num\_SBS}{total\_req\_num} \tag{21}$$

5.3 Evaluation results and discussions

5.3.1 Hit rate of different level

The performance of CCSOA is evaluated on the hit rate in the UE level and the hit rate in the SBS level in the case of one content and multiple contents. The parameters considered are average degree value, number of

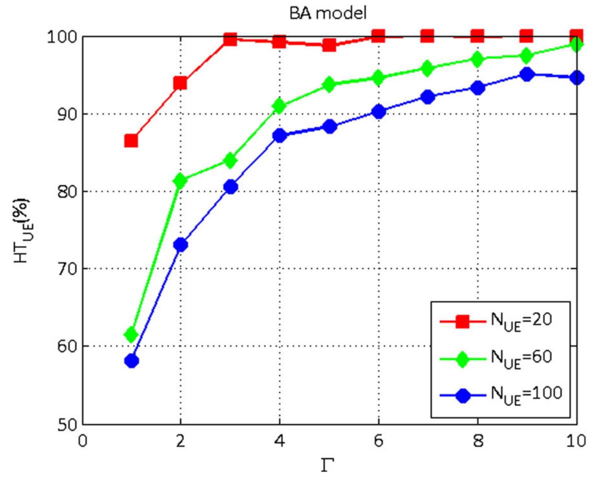

Fig. 12 Impact of number of nodes to hit rate in BA model ($\overline{d_{UE}} = 4$)

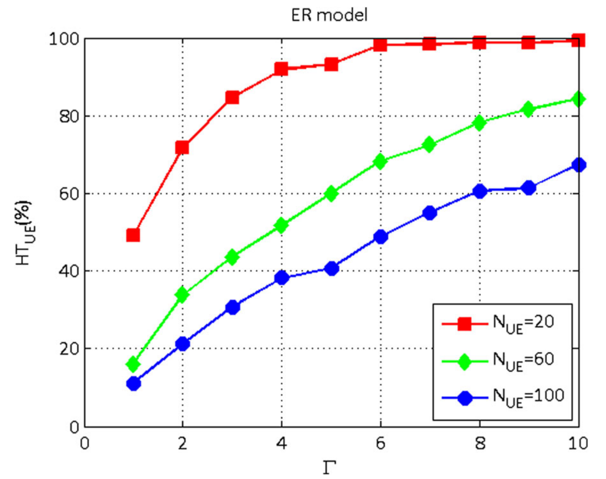

Fig. 13 Impact of nodes number to hit rate in the UE level in ER model ($\overline{d_{UE}} = 4$)

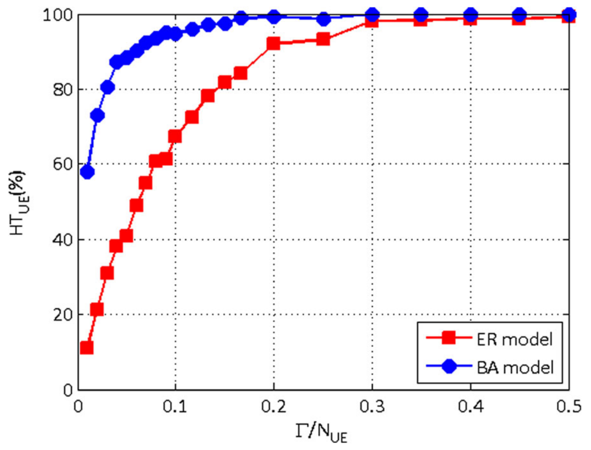

Fig. 14 Impact of percentage of content copies Γ/N_{UE} to hit rate in the UE level ($\overline{d_{UE}} = 4$)

nodes in the UE level, Γ/N_{UE}, Zipf parameter, and $^{T_{SBS}}/_{T_{UE}}$ as well as different network model. The performance of CCSOA on hit rate is compared with CEE and ProbCache.

5.3.1.1 The case of one content The content-centric slicing considering one content is analyzed in order to investigate the network slice for one content on cache resources and communication resources. The impact of Γ, N_{UE}, Γ/N_{UE}, and average degree as well as network models to the hit rate in the UE level are evaluated.

The results of CCSOA on the hit rate in the UE level in BA model are shown in Figs. 10 and 11 when the number of UEs is 100. The results of the hit rate in the UE level in ER model with the same number of UEs are shown in Figs. 12 and 13. The relations of the hit rate in the UE level among maximum content copies, different network model, average degree value, and the number of UEs are given. When average degree value, the number of UEs and maximum content copies are the same, hit rate in BA model is higher than that in ER model due to some UEs with very high degree value in BA model. Meanwhile, with the similar degree value of UEs in ER model, when the number of UEs is the same, then the higher the maximum content copies and the average degree value are, the higher the hit rate in the UE level is. The reason is that in a network with higher average degree value, the UEs can contact more UEs. Therefore, the possibility of getting the content is higher. In a network with larger number of UEs, more count content copies is needed in order to maintain the high enough hit rate in the UE level.

In BA model, for a network with N_{UE} = 100, the caching resource slicing for one content in the UE level is 6 copies when the average degree is 4, and the hit rate in the UE level is above 80%. However, in ER model, for a network with N_{UE} = 100, the caching resource slicing for one content in the UE level is 10 copies when the average degree is 4, the hit rate in the UE level is still under 80%.

The impact of the percentage of content copies (namely Γ/N_{UE}) to the hit rate in the UE level is shown in Fig. 14. When the percentage of content copies Γ/N_{UE} is greater, the hit rate in the UE level is higher. In addition, with the same percentage of Γ/N_{UE}, the hit rate in the UE level is higher in BA model than that in ER model.

5.3.1.2 The case of multiple contents In this section, the performance of CNSC-RAN exploiting CCSOA considering multiple contents is evaluated. The factors

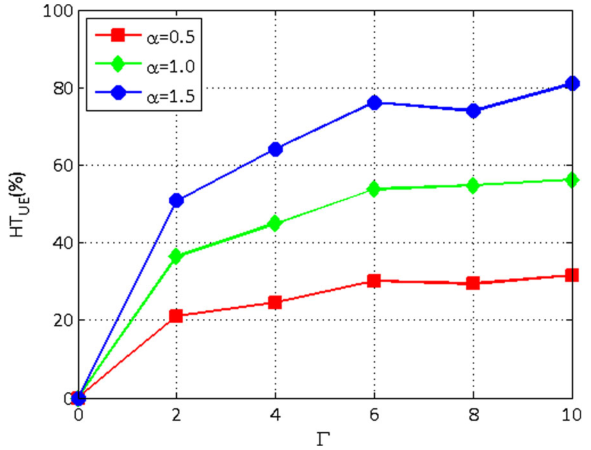

Fig. 15 Impact of Zipf parameter to hit rate in the UE level in ER model ($\overline{d_{UE}} = 4, N_{UE} = 60, ^{T_{SBS}}/_{T_{UE}} = 5$)

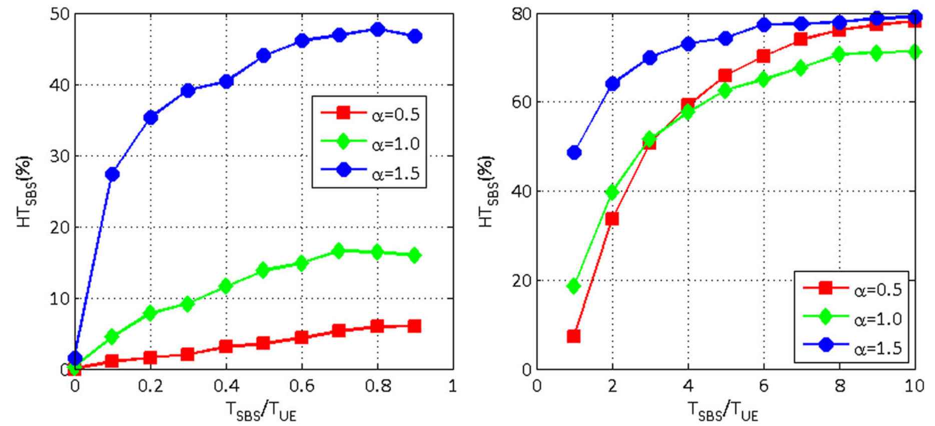

Fig. 16 Impact of ${}^{T_{\rm SBS}}/_{T_{\rm UE}}$ to hit rate in the SBS level in ER model ($\overline{d_{UE}} = 4, N_{UE} = 60, \Gamma = 6$)

affecting the hit rate in the UE level and SBS level are mainly the content popularity of the contents (Zipf parameter, namely α), network model (including ER model and BA model) which indicates the connection feature of UEs, the number of UEs in the UE level ($N_{\rm UE}$), the average degree, the maximum copies of one content, and the caching eviction period ratio (${}^{T_{\rm SBS}}/_{T_{\rm UE}}$). The hit rate of CCSOA is also compared with CEE and Prob-Cache in ER model.

The impact of Zipf parameter α to the hit rate in the UE level with ER model is shown in Fig. 15. It reveals that the more cache resources allocated to each content-centric slice is, namely the more the maximum copies of the content (Γ) cached in the UE level is, the more it would contribute to the hit rate in the UE level. At the same time, the greater Zipf parameter α is, the higher the hit rate in the UE level is.

The impact of Zipf parameter α and ${}^{T_{\rm SBS}}/_{T_{\rm UE}}$ to the hit rate in the SBS level with ER model is shown in Fig. 16. It reveals that when ${}^{T_{\rm SBS}}/_{T_{\rm UE}}$ increases, namely the time which contents are cached in SBS compared to that in the UE level is longer, the hit rate in the SBS level is higher. However, when $\alpha = 0.5$, the hit rate in the SBS level is only slightly the same when $\alpha = 1.0$ with ${}^{T_{\rm SBS}}/_{T_{\rm UE}} = 4$, which is also shown in Fig. 17. The reason is that one copy of the same content cached in the UE level is also cached in SBS, and the content can be cached for longer time in the SBS level. Therefore, the parameter ${}^{T_{\rm SBS}}/_{T_{\rm UE}}$ can be used for SBSs to cache unpopular contents for UEs in order to get high hit rate in the SBS level. In addition, when α becomes lower, more cache resources in SBS would be required to meet high hit rate in the SBS level.

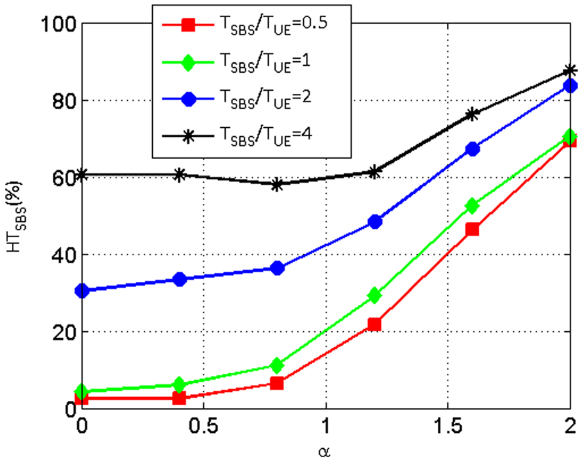

Fig. 17 Impact of ${}^{T_{\rm SBS}}/_{T_{\rm UE}}$ to hit rate in the SBS level in ER model ($\overline{d_{\rm UE}} = 4, N_{\rm UE} = 60, \Gamma = 6$)

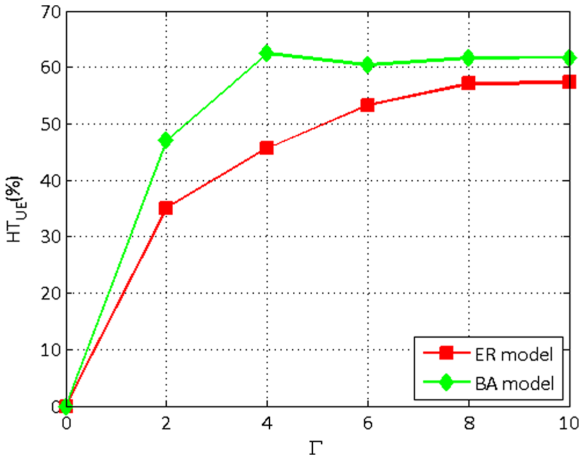

Fig. 18 Impact of network feature to the hit rate in the UE level ($\alpha = 1, \overline{d_{\rm UE}} = 4, N_{\rm UE} = 60, {}^{T_{\rm SBS}}/_{T_{\rm UE}} = 5$)

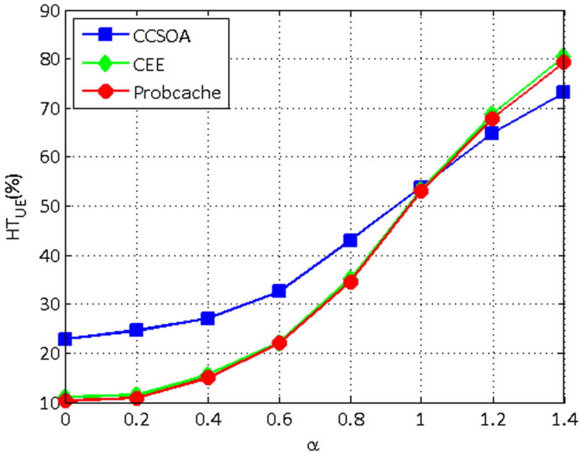

Fig. 19 Impact of Zipf parameter to hit rate in the UE level in ER model ($\overline{d_{UE}} = 4, N_{UE} = 60, \Gamma = 8, {}^{T_{SBS}}/_{T_{UE}} = 5$)

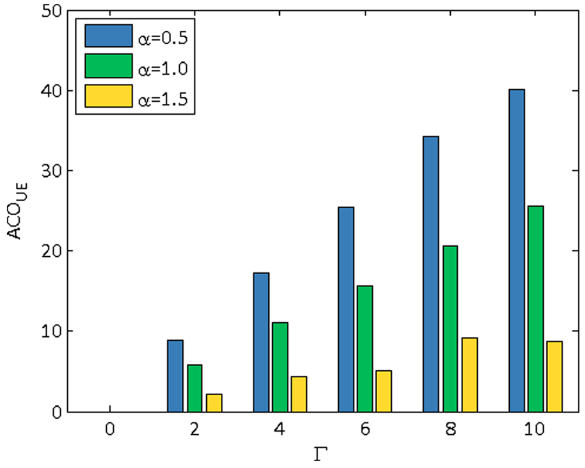

Fig. 21 Impact of Γ to ACO in the UE level in ER model ($\overline{d_{UE}} = 4, N_{UE} = 60, {}^{T_{SBS}}/_{T_{UE}} = 5$)

Since BA model emulates the connection feature of society network, while ER model is similar with the random UE interconnection, the impact of network model and the maximum copy of contents to the hit rate in the UE level is analyzed in Fig. 18. It is shown that the hit rate in BA model is greater than that in ER model when the maximum copies of one content Γ is the same.

The impact of Zipf parameter to the hit rate in the UE level in ER model is investigated in Fig. 19 by comparing CCSOA with CEE and ProbCache. From Fig. 19, when Zipf parameter is lower than 1.2, the proposed scheme CCSOA achieves the higher hit rate in the UE level. When Zipf parameter is greater than 1.2, the hit rate of CCSOA is a little lower than that of another two schemes; however, the hit rate in the UE level is more stable than that of CEE and ProbCache with different content popularity.

The impact of Zipf parameter to the hit rate in the SBS level of CCSOA compared with CEE and ProbCache in ER model is demonstrated in Fig. 20. It reveals that CCSOA achieves higher hit rate than that of CEE and ProbCache. When Zipf parameter is lower than 1.2, the hit rate in the SBS level of CCSOA changes slightly because more cache resources are allocated to keep the hit rate in the SBS level with CCSOA. When Zipf parameter is greater than 1.2, the hit rate in the SBS level of three schemes increases faster.

5.3.2 Average cache occupation

The performance of CCSOA on average cache occupation in the UE level and that in the SBS level are

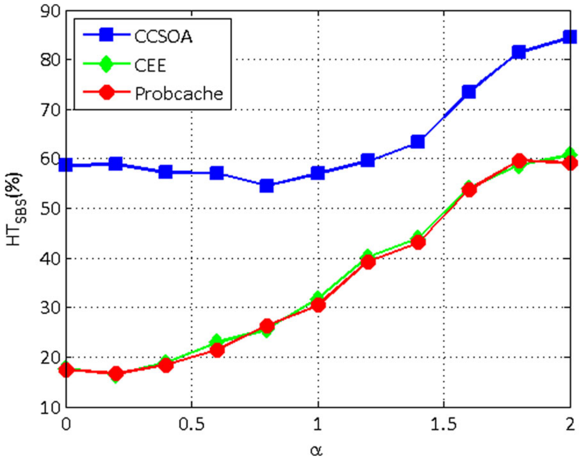

Fig. 20 Impact of Zipf parameter to hit rate in the SBS level in ER model ($\overline{d_{UE}} = 4, N_{UE} = 60, \Gamma = 8, {}^{T_{SBS}}/_{T_{UE}} = 5$)

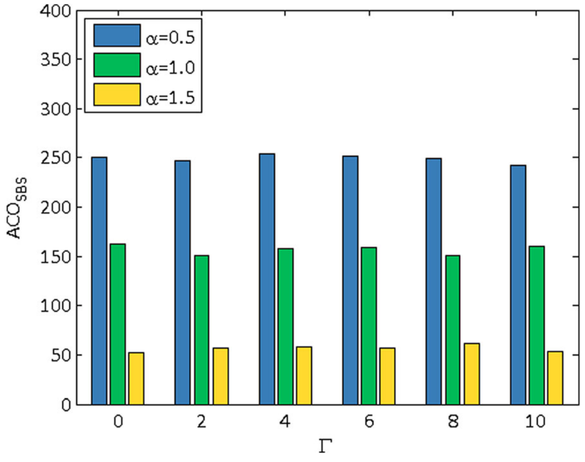

Fig. 22 Impact of Γ to ACO in the SBS level in ER model with different α ($\overline{d_{UE}} = 4, N_{UE} = 60, {}^{T_{SBS}}/_{T_{UE}} = 5$)

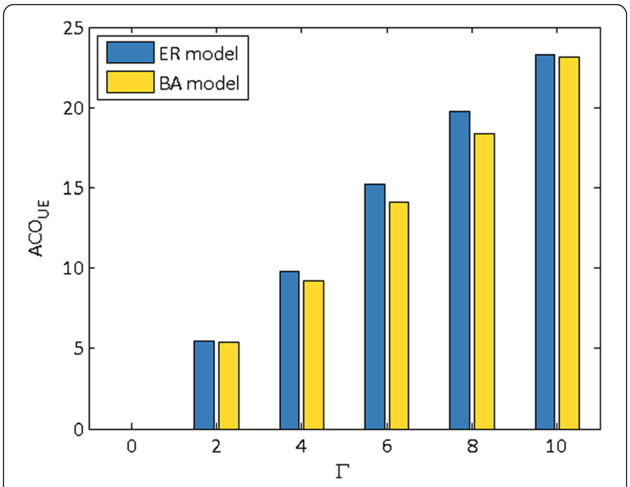

Fig. 23 Impact of Γ to ACO with different network models ($\alpha = 1, \overline{d_{UE}} = 4, N_{UE} = 60, {}^{T_{SBS}}/_{T_{UE}} = 5$)

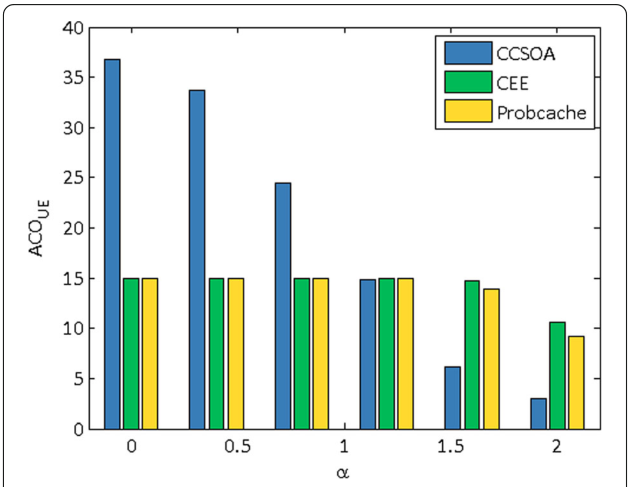

Fig. 25 Impact of Zipf parameter to ACO in the UE level in ER model ($\overline{d_{UE}} = 4, N_{\mathrm{UE}} = 60, \Gamma = 8, {}^{T_{\mathrm{SBS}}}/_{T_{\mathrm{UE}}} = 5$)

analyzed, the parameters considered are Γ, Zipf parameter, the network model, and ${}^{T_{\mathrm{SBS}}}/_{T_{\mathrm{UE}}}$. The average cache occupation of CCSOA is compared with CEE and ProbCache.

In Fig. 21, the impact of Γ and Zipf parameter to average cache occupation in the UE level is evaluated in ER model. It indicates that lower α leads to more cache occupation in the UE level. Once the maximum of content copies Γ in the UE level increases, more cache resources are sliced for contents in the UE level. When the number of N_{UE} is 60, the popularity of contents is high ($\alpha = 1.5$), and the maximum of content copies Γ is equal to or more than 8 copies in the UE level, that is to say, Γ/N_{UE} is more than 13%, no more cache occupation allocated in the UE level is required.

The impact of Γ to average cache occupation in the SBS level in ER model is shown in Fig. 22. From Fig. 22, it is visible that the parameter Γ of the UE level has little effect to the cache resources sliced in the SBS level, and high Zipf parameter α brings about less cache resource occupation in the SBS level.

Figure 23 shows the evaluation result on the impact of Γ to average cache occupation with different network models, the average cache occupation tends to nearly the same with different network models when Γ becomes great. The cache occupation is less in BA model than that in ER model because those nodes with great degree in BA model contributes more to the less average cache occupation.

The performance of CCSOA on the impact of ${}^{T_{\mathrm{SBS}}}/_{T_{\mathrm{UE}}}$ to average cache occupation in the SBS level

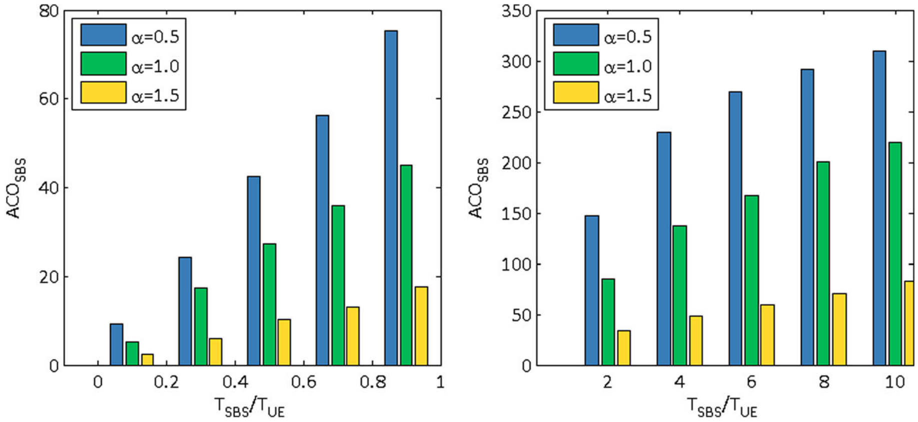

Fig. 24 Impact of ${}^{T_{SBS}}/_{T_{UE}}$ to ACO with different α in the SBS level ($\overline{d_{\mathrm{UE}}} = 4, N_{\mathrm{UE}} = 60, \Gamma = 6$)

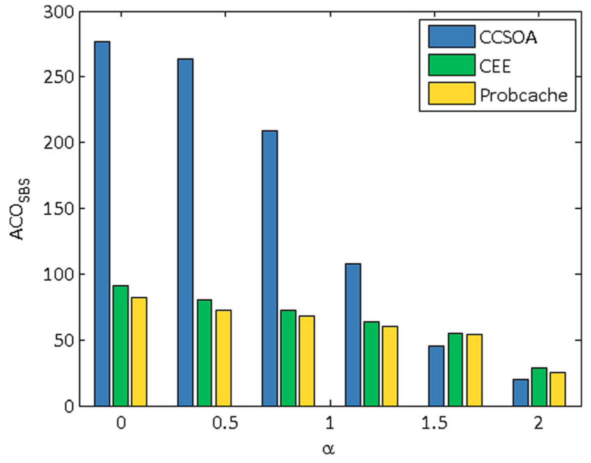

Fig. 26 Impact of Zipf parameter to ACO in the SBS level in ER model ($\overline{d_{\mathrm{UE}}} = 4, N_{\mathrm{UE}} = 60, \Gamma = 8, T_{\mathrm{SBS}}/T_{\mathrm{UE}} = 5$)

with different α in ER model is presented in Fig. 24. It reveals that $T_{\mathrm{SBS}}/T_{\mathrm{UE}}$ is positively correlated with average cache occupation in the SBS level, and the greater α is, the lower ACO is.

In Fig. 25, the performance of CCSOA is given on the impact of Zipf parameter to average cache occupation in the UE level in ER model compared with CEE and ProbCache. It indicates that the content cache slice allocated to UE level is more when α is lower than 1.2, while the cache slice is less when α is greater than 1.2.

In Fig. 26, the performance of CCSOA on the impact of Zipf parameter to average cache occupation is compared with CEE and ProbCache in the SBS level with ER model. It is shown that more cache resources are sliced in the SBS level in CCSOA than that in other baselines when α is small. But when α becomes greater, the cache resource occupation of CCSOA is lower than that of CEE and ProbCache.

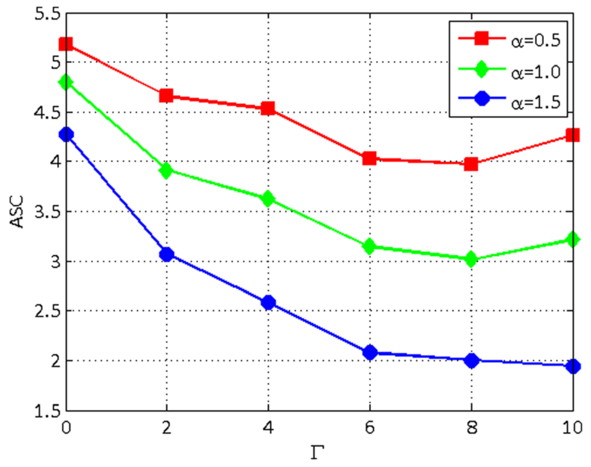

Fig. 27 Impact of Γ and Zipf parameter to ASC in ER model ($\overline{d_{\mathrm{UE}}} = 4, N_{\mathrm{UE}} = 60, T_{\mathrm{SBS}}/T_{\mathrm{UE}} = 5$)

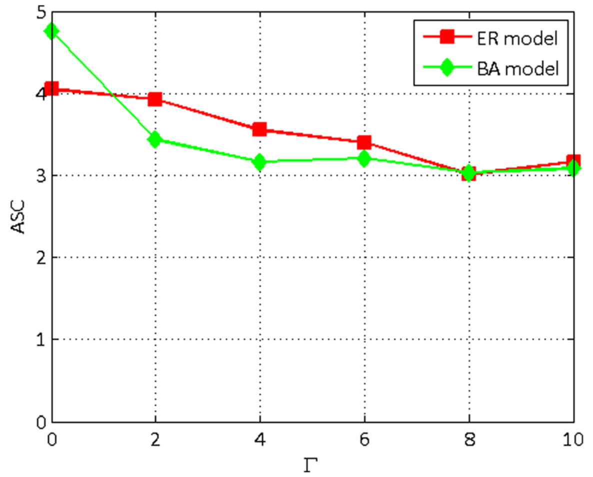

Fig. 28 Impact of network model and Γ to ASC ($\alpha = 1, \overline{d_{\mathrm{UE}}} = 4, N_{\mathrm{UE}} = 60, T_{\mathrm{SBS}}/T_{\mathrm{UE}} = 5$)

5.3.3 Average system cost

The average system cost of CCSOA is evaluated, which is the average weighted hop to get the contents for per content request. The parameters taking into consideration are Zipf parameter, Γ, network model, and $T_{\mathrm{SBS}}/T_{\mathrm{UE}}$. The average system cost of CCSOA is also compared with CEE and ProbCache in ER model.

The average system cost in ER model with different Zipf parameter and Γ is shown in Fig. 27. It reveals that more cache resources sliced for contents in the UE level contributes to less system cost, and the greater the Zipf parameter is, the less system cost it is to get the contents.

The impact of network model and Γ to the average system cost of CCSOA is demonstrated in Fig. 28. As Γ increases,the system cost decreases, and it is only a little lower in BA model than that in ER model with the same Γ, which is the maximum copies of content sliced for each content.

The impact of $T_{\mathrm{SBS}}/T_{\mathrm{UE}}$ and Zipf parameter to average system cost is shown in Fig. 29 with ER model. With the increase of $T_{\mathrm{SBS}}/T_{\mathrm{UE}}$, the average system cost becomes lower. That is to say, great $T_{\mathrm{SBS}}/T_{\mathrm{UE}}$ results in caching those unpopular contents in the SBS level for UEs. It also reveals that when $T_{\mathrm{SBS}}/T_{\mathrm{UE}}$ is great enough ($T_{\mathrm{SBS}}/T_{\mathrm{UE}} > 9$),

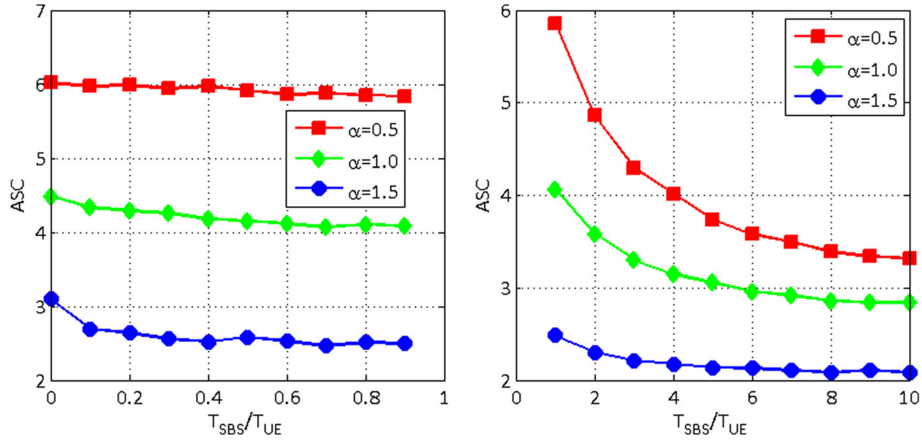

Fig. 29 Impact of $^{T_{\mathrm{SBS}}}/_{T_{\mathrm{UE}}}$ to ASC with different Zipf parameter ($\overline{d_{\mathrm{UE}}} = 4, N_{\mathrm{UE}} = 60, \Gamma = 6$)

the average system cost to get the content tends to be practically the same with a given α.

The performance of CCSOA on average system cost in ER model is compared with that of CEE and ProbCache in Fig. 30 when Zipf parameter is smaller than 1.5. The system cost which can be indicated as the communication resources allocated for transmitting contents is lower than another two schemes. When Zipf parameter is greater than 1.2, CCSOA has about one unit cost greater than other schemes. One unit cost is indicated as the cost of UE transmitting to UE. In other words, more contents are found in the third step in CCSOA, while more contents are found in step 1 or step 2. However, when Zipf parameter is greater than 1.2, CCSOA outperforms in the less cache occupation and high hit rate in the SBS level.

5.3.4 Request traffic reduction to MBS

The performance of request traffic reduction to MBS (RTR) of CCSOA is simulated to evaluate the request burden of MBS alleviated by SBS level and UE level, considering the influence of parameters including Zipf parameter α, Γ, network model as well as $^{T_{\mathrm{SBS}}}/_{T_{\mathrm{UE}}}$. The performance of RTR of CCSOA is evaluated comparing with CEE and ProbCache.

The impact of Zipf parameter α and Γ to RTR in ER model is shown in Fig. 31. It indicates that RTR is not related to Γ, which is the maximum copies of cached content sliced to each content in the UE level, while it depends on the popularity of contents because each content will be cached in the SBS level for some time. When $\alpha = 0.5$, the RTR is about 70%, in other words, about 70% of the requests can be hit in either SBS level or UE level.

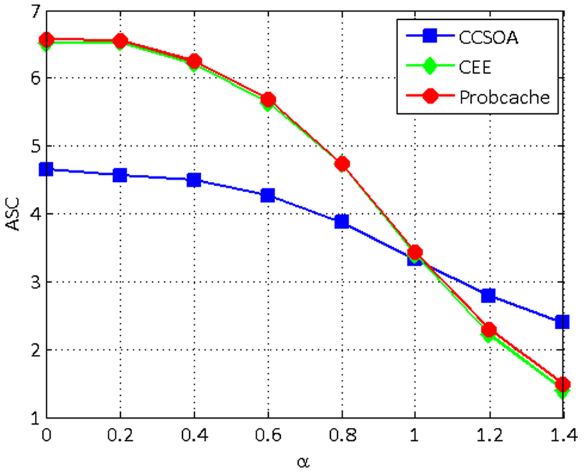

Fig. 30 The impact of α to ASC in ER model ($\overline{d_{\mathrm{UE}}} = 4, N_{\mathrm{UE}} = 60, \Gamma = 8, ^{T_{\mathrm{SBS}}}/_{T_{\mathrm{UE}}} = 5$)

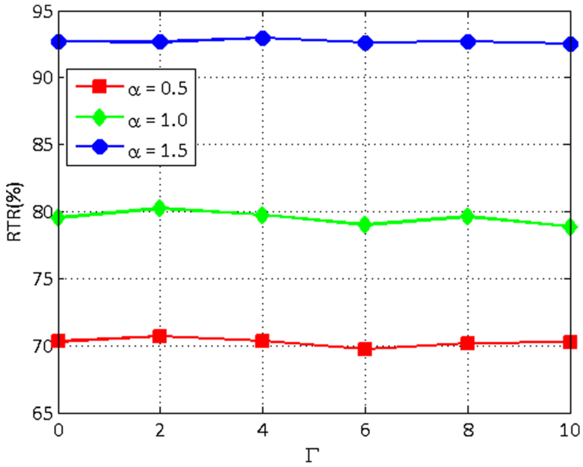

Fig. 31 The impact of α to RTR in ER model ($\overline{d_{\mathrm{UE}}} = 4, N_{\mathrm{UE}} = 60, ^{T_{\mathrm{SBS}}}/_{T_{\mathrm{UE}}} = 5$)

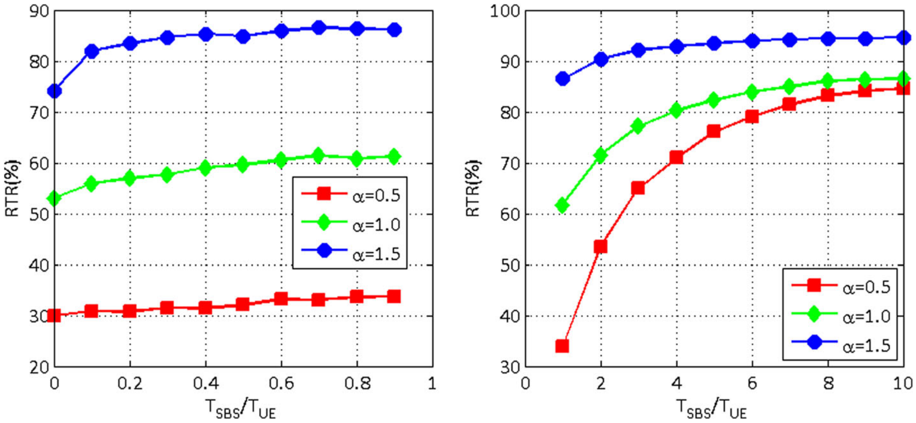

Fig. 32 The impact of $^{T_{SBS}}/_{T_{UE}}$ and Zipf parameter to RTR ($\overline{d_{UE}} = 4, N_{UE} = 60, \Gamma = 6$)

The impact of $^{T_{\mathrm{SBS}}}/_{T_{\mathrm{UE}}}$ and Zipf parameter to RTR in ER model is investigated in Fig. 32. It shows that greater $^{T_{\mathrm{SBS}}}/_{T_{\mathrm{UE}}}$ contributes to more request traffic reduction to MBS, which indicates that contents can be cached in the SBS level for longer time so that the request can be more probable to be hit in the edge of network. From Fig. 32, it is visible that with a given α, when $^{T_{\mathrm{SBS}}}/_{T_{\mathrm{UE}}} > 8$, the RTR tends to be smooth.

The simulation result on the impact of Γ to RTR with different network models is demonstrated in Fig. 33. It shows that RTR ranges around 79% when Γ increases from 2 to 8, that is to say, the feature of network and Γ give rise to less effect on RTR.

In Fig. 34, the impact of Zipf parameter to RTR is evaluated by comparing the RTR of CCSOA with CEE and ProbCache. Obviously, CCSOA outperforms when Zipf parameter is less than 1.2.

From the performance evaluation results above, compared with CEE and ProbCache scheme, it is shown that when $\overline{d_{\mathrm{UE}}} = 4, N_{\mathrm{UE}} = 60, \Gamma = 8, {}^{T_{\mathrm{SBS}}}/_{T_{\mathrm{UE}}} = 5$, the hit rate of CCSOA in the UE level and SBS level outperforms when $\alpha < 1$; the average cache occupation of CCSOA in the UE level and SBS level outperforms when $\alpha > 1.2$ and $\alpha > 1.5$ respectively; with ER model, the average system cost of CCSOA is less than that of CEE and ProbCache when $\alpha \leq 1.0$; the request traffic reduction of CCSOA is higher than that of CEE and ProbCache when $\alpha < 1.5$

6 Conclusions

In this paper, the network slicing and resource optimization on content in cache-enabled hybrid radio

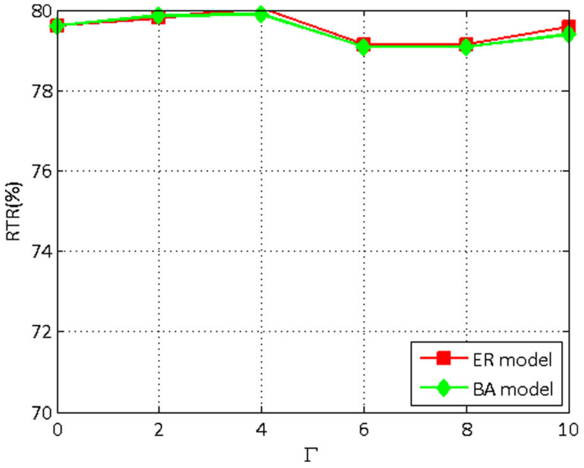

Fig. 33 The impact of Γ to RTR with different network models ($\alpha = 1, \overline{d_{UE}} = 4, N_{UE} = 60, {}^{T_{SBS}}/_{T_{UE}} = 5$)

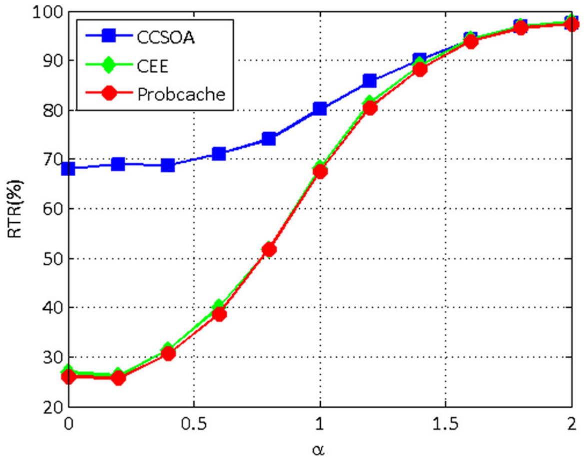

Fig. 34 The impact of Zipf parameter to RTR to MBS in ER model ($\overline{d_{UE}} = 4, N_{UE} = 60, \Gamma = 8, {}^{T_{SBS}}/_{T_{UE}} = 5$)

access network based on complex network are investigated. A cooperative caching framework named CNSC-RAN is proposed including MBS level, SBS level, and UE level. Based on CNSC-RAN, the functional modules including MBS controller, SBS controller, and UE controller are presented. Static and dynamic content-oriented network slicing procedure is illustrated. The process is designed to get the content required by UEs. In order to obtain the optimized resources sliced to each content in the framework, the content-oriented slicing is modeled and analyzed by using complex network and optimization theory and is formulated to minimize the average system cost to get the contents required by users in a known network architecture by using ER model and BA model. The problem is solved by a heuristic algorithm named CCSOA. The performance of CCSOA is evaluated by the metrics including hit rate, average cache occupation, and average system cost as well as request traffic reduction to MBS in dynamic content-oriented network slicing procedure enabling UEs with self-evicting contents. As future work, we plan to investigate the performance of network slicing optimization on content caching considering the effect of user mobility.

Acknowledgements
This work is supported in part by the National Natural Science Foundation of China under Grants 61471062, 61431008, and also by the State Major Science and Technology Special Projects under Grant 2017ZX03001014.

Authors' contributions
HJ contributed on the design of the methods on the content-oriented network slicing. HL presented the performance evaluation of CCSOA. CZ participated in the design and optimization of framework based on cache-enabled hybrid radio access network. All authors have read and approved the final manuscript.

Authors' information
Hao Jin received the PhD degree from Beijing University of Posts and Telecommunications in 1996. She is currently an associate professor in the Key Laboratory of Universal Wireless Communications for Ministry of Education, Beijing University of Posts and Telecommunications, China. Her research interests include optimization of mobile wireless communication networking and mobile edge computing.
Haiya Lu received the BEng degree from Nanjing University of Posts and Telecommunications in 2016. He is a graduate student for master degree in the Key Laboratory of Universal Wireless Communications for Ministry of Education, Beijing University of Posts and Telecommunications, China. His current research interests include mobile caching in wireless network and mobile cloud computing systems.
Chenglin Zhao received the PhD degree from Beijing University of Posts and Telecommunications in 1997. He is currently a professor in the Key Laboratory of Universal Wireless Communications for Ministry of Education, Beijing University of Posts and Telecommunications, China. His research interests include wireless resource management, cognitive radio network and wireless sensor network.

Competing interests
The authors declare that they have no competing interests.

Publisher's Note

Received: 31 July 2017 Accepted: 23 November 2017
Published online: 04 January 2018

References

1. Peng M, Zhang K. Recent Advances in Fog Radio Access Networks: Performance Analysis and Radio Resource Allocation [J]. IEEE Access, **4**(99), 5003–5009 (2016)
2. ETSI NFV ISG, Network functions virtualisation, white paper #3 [Online], October 2014. Available: https://portal.etsi.org/Portals/0/TBpages/NFV/Docs/NFV_White_Paper3.pdf
3. S Abdelwahab, B Hamdaoui, M Guizani, et al., Network function virtualization in 5G [J]. IEEE Commun. Mag. **54**(4), 84–91 (2016)
4. C Rotsos, D King, A Farshad, et al., Network service orchestration standardization: a technology survey [J]. Comput. Stand. Interfaces. **54**, 203–215 (2017)
5. P Demestichas, A Georgakopoulos, D Karvounas, et al., 5G on the horizon: key challenges for the radio-access network [J]. IEEE Veh. Technol. Mag. **8**(3), 47–53 (2013)
6. ID Silva, G Mildh, A Kaloxylos, et al., Impact of network slicing on 5G Radio Access Networks [C]. European Conference on Networks and Communications (2016), p.153–157
7. X Foukas, G Patounas, A Elmokashfi, et al., Network slicing in 5G: survey and challenges [J]. IEEE Commun. Mag. **55**(5), 94–100 (2017)
8. M Richart, J Baliosian, J Serrat, et al., Resource slicing in virtual wireless networks: a survey [J]. IEEE Trans. Netw. Serv. Manag. 13(3), 462–476 (2016)
9. Q Liang, X Cheng, SCH Huang, D Chen, Opportunistic sensing in wireless sensor networks: theory and application [J]. IEEE Trans. Comput. **63**(8), 2002–2010 (2014)
10. B Li, S Li, A Nallanathan, C Zhao, Deep sensing for future spectrum and location awareness 5G communications [J]. IEEE J. Sel. Areas Commun. **33**(7), 1331–1344 (2015)
11. B Li, S Li, A Nallanathan, Y Nan, C Zhao, Z Zheng, Deep sensing for next-generation dynamic spectrum sharing: more than detecting the occupancy state of primary spectrum [J]. IEEE Trans. Commun. **63**(7), 2442–2457 (2015)
12. X Zhou, R Li, T Chen, et al., Network slicing as a service: enabling enterprises' own software-defined cellular networks [J]. IEEE Commun. Mag. **54**(7), 146–153 (2016)
13. Y Liu, JC Point, KV Katsaros, et al., SDN/NFV based caching solution for future mobile network (5G) [C]. European Conference on Networks and Communications. (IEEE, 2017), pp. 1–5
14. S Vassilaras, L Gkatzikis, N Liakopoulos, et al., The algorithmic aspects of network slicing [J]. IEEE Commun. Mag. **55**(8), 112–119 (2017)
15. K Katsalis, N Nikaein, E Schiller, et al., 5G Architectural Design Patterns [C]. IEEE International Conference on Communications Workshops. (IEEE, 2016), pp. 32–37
16. Devlic A, Hamidian A, Liang D, et al. NESMO: Network slicing management and orchestration framework [C]. IEEE International Conference on Communications Workshops. (IEEE, 2017)
17. C Liang, FR Yu, X Zhang, Information-centric network function virtualization over 5g mobile wireless networks [J]. Netw IEEE **29**(3), 68–74 (2015)
18. R Huo, FR Yu, T Huang, et al., Software defined networking, caching, and computing for green wireless networks [J]. IEEE Commun. Mag. 54(11), 185–193 (2016)
19. Y Wang, D Lin, C Li, et al., Application driven network: providing on-demand services for applications [C]. Conference on ACM SIGCOMM 2016 Conference. (ACM, 2016), pp.617–618
20. A Nakao, P Du, Application-specific slicing for MVNO and traffic characterization [Invited] [J]. IEEE/OSA J. Opt. Commun. Networking **9**(2), 256–262 (2017)
21. X Li, R Casellas, G Landi, et al., 5G-Crosshaul network slicing: enabling multi-tenancy in mobile transport networks [J]. IEEE Commun. Mag. 55(8), 128–137 (2017)
22. K Katsalis, N Nikaein, A Edmonds, Multi-Domain Orchestration for NFV: Challenges and Research Directions [C]. International Conference on Ubiquitous Computing and Communications and 2016 International Symposium on Cyberspace and Security. (IEEE, 2017), pp. 189–195
23. H Jin, D Xu, C Zhao, et al., Information-centric mobile caching network frameworks and caching optimization: a survey [J]. EURASIP J. Wirel. Commun. Netw. 2017(1), 1–33 (2017)
24. Z Hu, Z Zheng, T Wang, et al., Caching as a service: small-cell caching mechanism design for service providers [J]. IEEE Trans. Wirel. Commun. **15**(10), 6992–7004 (2016)

25. X Li, X Wang, C Zhu, et al., Caching-as-a-Service: virtual caching framework in the cloud-based mobile networks [C]. Computer Communications Workshops. (IEEE, 2015), pp 372–377
26. K Wang, FR Yu, H Li, et al. Information-Centric Wireless Networks with Virtualization and D2D Communications [J]. IEEE Wireless Communications, **99**, 104–111 (2016)
27. Ying Loong Lee; Jonathan Loo; Teong Chee Chuah. A New Network Slicing Framework for Multi-Tenant Heterogeneous Cloud Radio Access Networks [C]. 2016 International Conference on Advances in Electrical, Electronic and System Engineering, (2016), pp. 414–420
28. C Vlachos, V Friderikos, Optimal virtualized resource slicing for device-to-device communications [C]. IEEE Global Communications Conference. (IEEE, 2015), pp. 1–7
29. R Kokku, R Mahindra, H Zhang, et al., NVS: a substrate for virtualizing wireless resources in cellular networks [J]. IEEE/ACM Trans. Networking 20(5), 1333–1346 (2012)
30. R Riggio, A Bradai, D Harutyunyan, et al., Scheduling wireless virtual networks functions [J]. IEEE Trans. Netw. Serv. Manag. 13(2), 240–252 (2016)
31. C Ghribi, M Mechtri, O Soualah, et al., SFC Provisioning over NFV Enabled Clouds [C]. IEEE, International Conference on Cloud Computing. (IEEE, 2017), pp. 423–430
32. Y Zhou, FR Yu, J Chen, et al., Resource allocation for information-centric virtualized heterogeneous networks with in-network caching and mobile edge computing [J]. IEEE Trans. Veh. Technol. 66(12), 11339–11351 (2017)
33. C Liang, FR Yu, Enhancing mobile edge caching with bandwidth provisioning in software-defined mobile networks [C]. IEEE International Conference on Communications. (IEEE, 2017), pp. 1–6
34. S Retal, M Bagaa, T Taleb, et al., Content delivery network slicing: QoE and cost awareness [C]. IEEE International Conference on Communications. (IEEE, 2017), pp. 1–6
35. Zhou B, Cui Y, Tao M. Stochastic content-centric multicast scheduling for cache-enabled heterogeneous cellular networks [C]. ACM CONEXT 2015 Content and Delivery in WirelessNetworks. (ACM, 2015) , pp. 1–14
36. WK Chai, D He, I Psaras, Cache less for more in information-centric networks [C]. International Ifip Tc 6 Conference on NETWORKING, (2012), pp. 27–40
37. D Xu, D Fang, S Tang, et al., Content caching with virtual spatial locality in cellular network [J]. Pervasive Mob. Comput. **41** 365–380, (2017)
38. M. E. J. Newman, Networks: an introduction, (Oxford University Press, 2010)