



UWL REPOSITORY

repository.uwl.ac.uk

Cache space efficient caching scheme for content-centric mobile ad hoc networks

Zhang, Tiankui, Xu, Xiaogeng, Zhou, Le, Jiang, Xinwei and Loo, Jonathan ORCID:
<https://orcid.org/0000-0002-2197-8126> (2019) Cache space efficient caching scheme for content-centric mobile ad hoc networks. IEEE Systems Journal, 13 (1). pp. 530-541. ISSN 1932-8184

<http://dx.doi.org/10.1109/JSYST.2018.2851394>

This is the Accepted Version of the final output.

UWL repository link: <https://repository.uwl.ac.uk/id/eprint/6101/>

Alternative formats: If you require this document in an alternative format, please contact:
open.research@uwl.ac.uk

Copyright:

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy: If you believe that this document breaches copyright, please contact us at open.research@uwl.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

Cache Space Efficient Caching Scheme for Content Centric Mobile Ad Hoc Networks

Tiankui Zhang, *Senior Member, IEEE*, Xiaogeng Xu, Le Zhou, Xinwei Jiang and Jonathan Loo, *Member, IEEE*

Abstract—Content centric mobile ad hoc networks (CCMAN) have been demonstrated as a potential complementary to multimedia content dissemination in future wireless networks. In-network caching is one of the key techniques in CCMAN, which can significantly reduce the network traffic load and improve the content retrieval performance. However, the challenges for caching in CCMAN are, all the wireless nodes can act as users and routers simultaneously, and the caching capability of each wireless node is limited by the cache size. So caching strategies should improve not only the caching performance but also the cache space efficiency. In this paper, we study the cache space efficient caching in CCMAN. Firstly, the theoretical performance of caching in CCMAN is analyzed, the cache utility is defined and derived. Then a cache space efficient caching (CSEC) scheme is proposed to improve the efficiency of cache space utilization in CCMAN. Accordingly, the analytical performance of the proposed CSEC in terms of the cache utility of CCMAN is given. The network performance of the proposed CSEC are evaluated by simulation. Simulation results show that the proposed CSEC can achieve optimal performance on cache utility, which can provide a tradeoff between cache hit ratio performance and occupied cache space.

Index Terms—Caching, content centric networking, dominating set, mobile ad hoc network.

I. INTRODUCTION

THE Future and emerging communication networks will pose extreme communication requirements such as very high throughput rates and very low latency, particularly in quality of experience (QoE) sensitive applications such as multimedia content delivery. Currently, a new network architecture named information centric networking (ICN) has been proposed as a technique that has the potential to reduce network traffic and content latency for multimedia content delivery [1].

ICN is a revolutionary network architecture for solving the content delivery problem of today's internet with TCP/IP [1]–[3]. Among several proposed architectures for ICN, content centric networking (CCN) architecture has attracted a great deal of attention, with the following properties: (a) in CCN, content is addressed by its name itself, the routing of the content retrieval is determined by the content name rather than

by the IP address; and (b) by content naming, content can be cached in intermediate routers. Based on the above design, nodes in network that own the contents can issue the data packet responding for the content requests.

Ad hoc networking with the advantages of fast deployment and easy device replacement has drawn more and more attention, which can provide various services such as communication, storage, and computing for a range of applications. Mobile ad hoc networks (MANET) can be used in rural or emergency scenarios that lack of infrastructure coverage. Since the MANET is self-organized, the wireless nodes may join and quit the network randomly, thus the communication connectivity is short-lived, and the network topology formed with the short-lived links varies dynamically. In such kind of network, when nodes are moving out of the network, the contents may vanish if the contents are delivered based on TCP/IP, but can still be remained in CCN based networks.

Content centric mobile ad hoc networks (CCMAN) have been demonstrated as a potential complementary to content dissemination in wireless networks [4], since CCN supports node mobility inherently, and can cope well with intermittent, short-lived connectivity, and dynamic topologies [5]. Therefore, the feasibility of applying CCN in wireless ad hoc scenarios, such as MANET [6], VANET [7], DTN [8] and WSN [9], has been recently discussed in the literature.

Since in-network caching is one of the key techniques in CCN, the caching strategy should be designed delicately. Most of the caching strategies proposed in the literature are designed in wired network scenarios, in which, the users are located at the edge of the networks and the routers are used to forward data packets and cache content replicas [10].

However, MANET are made up by terminals which should act as routers to achieve multi-hop communication between wireless nodes. Therefore, the assumption in wired networks that users located at the edge of the network is not feasible, and the caching strategies that push the contents to the network edges no longer applies. Additionally, the cache space of nodes in MANET is usually limited. All the features mentioned above make the existing caching placement strategies less effective when migrated to MANET. The in-networking caching of CCMAN is different from that of wired networks, in the following aspects:

- Users can be located at any place in the networks, not always in the edge of the networks, so the contents should be cached at strategic locations in the networks arguably and justifiably good for content delivery;
- Wireless nodes have limited capability on cache and computation, so the cache capability of wireless nodes

Manuscript received August 5, 2017; revised January 30 and May 9, 2018; accepted June 23, 2018. This work was supported by NSF of China (61502046), the 111 project (NO.B17007), and Director Funds of Beijing Key Laboratory of Network System Architecture and Convergence (2017BKL-NSAC-ZJ-07). (Corresponding author: Tiankui Zhang.)

T. Zhang, X. Xu, Z. Zhou, and X. Jiang are with the School of Information and Communication Engineering, Beijing Laboratory of Advanced Information Networks, Beijing University of Posts and Telecommunications, Beijing, 100876, China (e-mail: {zhangtiankui, xuxiaogeng, zhoule, jxw}@bupt.edu.cn).

Jonathan Loo is with the School of Computing and Engineering, University of West London, United Kingdom (e-mail: jonathan.loo@uwl.ac.uk).

should be fully explored for content caching efficiency improvement.

The concept of dominating set (DS) was first introduced into the wired CCN [11]. By adopting DS, an arbitrary flat topology of wired CCN was converted into a two-level hierarchy [11]. After that, a caching mechanism combined with content request routing, based on a virtual backbone network constructed by DS, was proposed. In the scenario of CCMAN, our previous work [12] extended the concept of DS as the generalized dominating set (GDS) to select some key wireless nodes for content caching, which made any two wireless nodes can communicate within a predefined number of hops. In [12], a cooperative caching placement scheme named GDSC was proposed based on GDS to minimize the network power consumption in CCMAN. Based on [12], Y. Liu et. al, who was one of the co-authors in [12], had led a variant study in [13], in which they proposed a cache replacement scheme, named as CoopCache, considering the local content popularity.

The existing studies mentioned above do not consider the cache space efficiency of wireless nodes in caching scheme designing, neither analyze the theoretical performance of caching with ubiquitous content request in CCMAN. In this paper, we study the theoretical performance of in-network caching in CCMAN and propose a cache space efficient caching (CSEC) scheme based on GDS method. Different from the works in [12] and [13], the cache space cost is taken into consideration to design the caching placement scheme, aiming to improve the content delivery performance on the cache hit ratio, average round trip hops (ARTH), and average round trip times (ARTT). The main contributions of this paper are following:

1) A theoretical analysis of in-network caching is formulated to provide the analytical performance of CCMAN. Different from the existing analytical models, the proposed model in this paper consider the feature that the content requests at a wireless node consist of the content requests generated by itself and the missed content request from its neighbor nodes in CCMAN. Here, the cache utility is defined and derived in terms of the cache hit probability and the content request arrival rate, which helps us understand the caching behaviors of CCMAN.

2) A CSEC scheme is proposed to improve the efficiency of cache space utilization in CCMAN. In CSEC, the method of GDS is used to select the proper nodes as the caching locations for content replicas, so that the cached contents can be fully utilized. Compared with the existing studies, the proposed scheme can improve both the caching performance and the cache space efficiency, which can provide a tradeoff between cache hit ratio performance and the occupied cache space.

3) An analytical cache model of the proposed CSEC with ubiquitous content request routing is derived based on the theoretical analyses of CCMAN caching. This model is used to study the performance of CSEC in terms of the cache utility. In addition, a simulation model of the proposed CSEC based on the analytical cache model is implemented and simulated using an open source simulation tool, ndnSIM.

4) The performance of the proposed CSEC in some random

network topologies is evaluated in the ndnSIM simulation environment. The simulation results on cache utility, ARTH gain, and ARTT gain of the proposed CSEC are obtained. Compared with two DS based caching schemes, GDSC in [12] and CoopCache in [13], the proposed CSEC can obtain the best caching performance in the scenarios of both the stable and moving networks.

This paper is arranged as follows: Section II introduces the related works about CCN caching in the Internet and wireless networks. The network model of CCMAN is given in Section III. Section IV studies the analytical caching performance of CCMAN. Section V is the proposed CSEC scheme and its analytical performance. In Section VI, the simulation results are shown. Finally, this paper is concluded. The main symbols and variables used in this paper are summarized in Table I.

TABLE I: Symbols and Variable List

Parameter	Description
N	Number of nodes
R	Number of contents
v_i	The i^{th} node
r	The r^{th} content
λ_i	Content request arrival rate of node v_i
g_i	Cache frequency of node v_i
p_i^c	Cache probability of node v_i
p_i^h	Cache hit probability node v_i
p_i^m	Cache miss probability node v_i
q_i	Number of neighbor nodes of node v_i
p_i^r	Content replacement probability of node v_i
h	Cache hit ratio
η	Cache utility

II. RELATED WORKS

A. CCN Caching in Internet

The caching management problems include the caching placement and caching replacement [12]. The caching placement is a problem that focuses on “when” and “where” to cache the contents and caching replacement pays attention to “which” replicas should be sacrificed to cache the newly coming content when the content store is overflow [12].

In the wired CCN, the most typical caching placement scheme and caching replacement scheme are leave copy everywhere (LCE) and least recently used (LRU) respectively, which had been widely used in web caching [14]. Although these caching strategies can be easily implemented, but the potential gain of caching is not fully utilized.

Some studies designed the caching placement schemes taking the different influence factors of caching into consideration. In [15], the caching placement decision was made according to the cache space size. The authors in [16] proposed to place the content replicas at the nodes with higher betweenness centrality, which was a network layer metric. From the view of the application layer influences, the caching placement decision can be made depending on the social relationships [17], the user preference [18], and the content popularity [19].

In terms of the caching implementation, the caching placement schemes can be sorted as cooperative caching and non-cooperative caching. cooperative caching is used to improve the efficiency of CCN caching. The caching strategies which

need information of all nodes in the networks belong to the explicit cooperative caching strategies [11], [20]. On the other hand, the implicit cooperative caching strategies need no or little interaction between nodes to achieve cooperative caching [21], [22].

As for non-cooperative caching, probability caching can be seen as one of the typical non-cooperative caching placement schemes. [23] presented a hop-based probabilistic caching. ProbCache proposed in [24] approximated the caching capability of a path and cached content replicas probabilistically to reduce the cache redundancy and improve the efficiency of the available cache resources along a content delivery path. Since many nodes caching the popularity contents forcefully damaged the diversity of cache, a content popularity and node ability matching based probability caching strategy was proposed in [26], which aimed to improve the content diversity.

B. CCN Caching in Wireless Networks

In this paper, we focus on the caching placement schemes in CCMAN. Before the CCN caching, several caching strategies in wireless ad hoc networks had been investigated in the literature. In particular, the solution in [27] aimed at caching the same content farther than a given number of hops. The concept of caching different content within a neighborhood was also exploited in [28], where nodes with similar interests and mobility patterns are grouped together to improve the cache hit ratio. In [29], distributed caching strategies for ad hoc networks were proposed either made nodes cache highly popular contents that passes by or recorded the data forwarding path to redirect future requests. [30] proposed a caching decision strategy based on the awareness of the contents that stored in neighbor users, aiming for differentiating the caching contents of each others. The content caching had been studied in the intermittently connected MANET as well, a cooperative cache-based content delivery framework was proposed in [31].

Recently, the studies on caching in CCMAN has been carried in [12], [13], [32], [33]. In [32], a content management architecture was proposed for CCMAN, which bounded the data and its physical location, and cached popular contents more intensively. In [33], a cooperative caching method was proposed based on the broadcasting nature of radio channel to improve the cooperative cache gain.

The concept of CCN has been introduced in VANET [34], [35]. [34] explored the social cooperation scheme among vehicles in the VANET and proposed the cooperative caching based on the auxiliary of partnership and messenger to enhance the CCN caching performance, to improve the user QoE on the multimedia streaming service. Considering the sufficient energy and cache space of vehicles, data transmission service in the VANET was proposed in [35] to push the data to vehicles for proactive caching.

In the light of recent advances, edge caching in mobile cellular networks has been investigated in [36]. A detailed survey of the literature on this topic is beyond the scope of this paper. Interested readers may refer to [36] and references therein.

C. Analytical Model of Caching

Besides the caching strategy design, the analytical cache model has been studied as well [37]–[43].

The authors in [37] proposed an analytical model based on the Markov model. In the proposed model, the single router system and the approximate model of two routers system were given, in which, the content requests from intermediate nodes were taken into consideration. An analytical model of ICN storage and bandwidth sharing was proposed in [38] based on independent reference model (IRM), in which the expressions of cache miss rate, round trip times (RTT) of content acquisition were given with LCE caching placement strategy and LRU caching replacement strategy. The authors in [39] proposed a caching placement strategy and its corresponding analytical model. The performance of LRU under the Shot Noise Model (SNM) model was analyzed in [40]. A recent study [41] analyzed the SNM model and gives the optimal policy for joint caching and popularity estimation. However, these analytical cache models mentioned above are designed in simple network topologies, such as the line or complete binary tree topologies.

The analytical cache model and network performance of CCN in a more general network topology has been studied in [42], [43]. The authors in [42] proposed a model for a general network and obtained the performance of RTT and throughput. A recent study [43] analyzed the network throughput and availability, assuming link failures in addition to content delivery delay. Both of these works are based on the work in [37], which obtained the multi-cache approximation by repetitively calculating single-cache in the network.

In CCMAN, the nodes act as both users and routers simultaneously, so the content requests arriving at a node consist of not only the content requests generated by itself but also the missed content requests from its neighbor nodes. Therefore, the existing caching analytical models cannot be extended into CCMAN directly. In this case, it is necessary to derive an analytical model to measure the influence of caching in CCMAN theoretically, that is one of the objectives of our work in this paper.

III. NETWORK MODEL OF CCMAN

In this section, we focus on CCMAN with an emphasis on caching. The work flow of CCMAN is given, and the key problem of the content caching in CCMAN is described.

In an arbitrary topology of CCMAN as shown in Fig. 1, all wireless nodes act as both users and routers. Hence, the nodes that are not directly connected can be interconnected via multi-hop communications.

The work flow of CCMAN is inherited from the traditional CCN as depicted in Fig. 2. There are two types of packets in CCMAN, interest packet carrying the requested content name, and data packet carrying the content data. Three data structures are defined in CCMAN: forwarding information base (FIB) for guiding the forwarding of the interest packet, pending interest table (PIT) for recording the forwarded interest packet without responding and content store (CS) for content caching. Interest packets are issued by the users (termed as the content consumers in CCN) to request their concerned contents. When

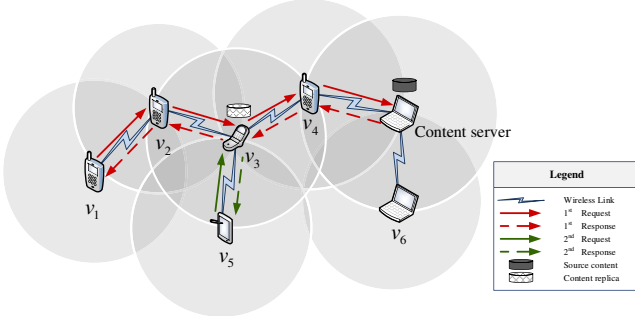


Fig 1. Example of CCMAN

an interest packet arrives at a wireless node, a longest-prefix match lookup is done on its content name, following the sequence of CS, PIT and FIB successively. If there is a matching data in the CS, which is defined as a “cache hit”, it will be sent back along the reverse path of the interest packet, and the interest packet will be discarded. Otherwise, if there is a matching PIT entry, the incoming face will be added to it and the interest packet will be discarded as well. Finally, if no matching PIT entry exists, the interest packet will be forwarded to the next hop according to the FIB, and a new PIT entry will be added in order to direct the data response route.

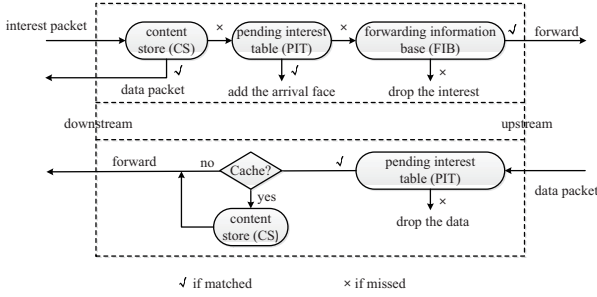


Fig 2. Work flow of CCMAN

Take the network topology of CCMAN in Fig. 1 for an example. The content server broadcasts the content items which are reachable to the entire network, where all nodes will update their FIB and forward the broadcast message. In Fig. 1, the node v_1 receives this broadcast message from node v_2 , then adds the corresponding content name and faces to its FIB. Once node v_1 requests this content, it generates an interest packet and then issues the interest packet to its neighbor nodes according to a specific routing, via node v_2 in this example. Interest packet is forwarded between nodes by multiple hops until there is a hit on CS of a node or the interest packet reaches the content server. So in this example, node v_1 obtains this content from the content server since node v_1 generates the first request to this content. Then, corresponding data packet is generated by the content server and forwarded along the reverse path of the interest packet by multi-hop as well. When the content goes through the node v_3 , a content replica is stored in its CS. After that, when the node v_5 sends an interest packet requesting the same content, node v_3 can response this content request and deliver the same content to node v_5 directly.

In the self-organized CCMAN, all the wireless nodes act as both the routers and the end users simultaneously. In such ad hoc scenario, the content requests can be generated by any node at anywhere in the networks. The performance gain of the content caching is affected by the content replicas placement. It is necessary to analyze the theoretical performance of content caching in the arbitrary topology scenario of CCMAN, and consider the content caching placement problem with limited cache size of nodes.

IV. THEORETICAL ANALYSIS OF CACHING PERFORMANCE

In this section, we analyze theoretical performance of caching in CCMAN, where the content requirements can come from any nodes in the network, and the content replicas can be cached anywhere in the network.

At the beginning of the analysis, we make the following definitions.

Definition 1. Cache frequency of a node indicates the number of different contents cached by the node per unit time. Cache frequency of node v_i is denoted as g_i .

Definition 2. Cache probability of a node represents the probability of the node to cache the content replicas carried by the passing data packets. The cache probability of node v_i is denoted as p_i^c .

Definition 3. Cache hit probability at a node denotes that when receiving an interest packet, the probability that there is a matched replica of the content in its CS. The cache hit probability of node v_i is p_i^h .

Definition 4. Cache miss probability at a node denotes that when receiving an interest packet, the probability of there is not a matched replica of the content in its CS. Denote the cache miss probability of node v_i as p_i^m , and $p_i^m = 1 - p_i^h$.

Definition 5. Cache hit ratio means the ratio of the number of interest packets that has been satisfied by the caching nodes rather than content server to the total number of interest packets in the networks.

Definition 6. Cache utility is defined as the ratio of cache hit ratio and the occupied cache space in the entire network.

In CCMAN, any node is able to cache the content replicas according to the defined caching placement strategy. We use the LRU caching replacement strategy. Since the cache space is limited compared with the vast amount of contents in the networks.

It is obvious that larger cache hit would improve the content deliver efficiency. However, CCMAN nodes normally have limited cache capacity due to cost concern, so we employ the cache utility to measure the caching performance of CCMAN. It means that, in CCMAN, we do not only measure the cache hit ratio as the wired CCN, but also study the cost of cache space occupation. The optimal cache utility can provide the best tradeoff between cache hit ratio and the cache space cost.

Hereinafter, we analyze the cache hit ratio and cache utility in terms of the cache probability with LRU caching replacement strategy.

Firstly, we prove that the content request arrival process at each node follows the Poisson process if the content request generating event of each node follows the Poisson process. Then we derive the cache hit probability function based on

the relationship between the cache frequency and the cache probability. Finally, we give the cache utility expression in terms of the content request generating rate and the cache hit probability.

We analyze the cache hit ratio of the network under the condition that node v_i cache contents carried by the pass by data packets with probability p_i^c in the following part of this section, and find out the impact factors of cache hit ratio.

Assuming that the content request generating process X of each node follows a Poisson process with rate λ , namely, $X \sim \pi(\lambda)$.

The analysis model of content caching of CCMAN is given in Fig. 3. In this model, the content request arrival rate at each node consists of two parts, content request generated by itself and missed content request from its neighbor nodes.

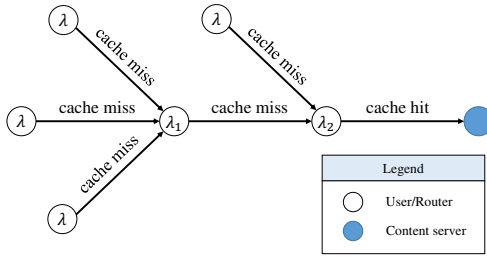


Fig 3. Analysis model of content caching of CCMAN

Lemma 1. The content request arrival process at each node follows Poisson process, namely, the event Y that node v_i issues interest packet follows Poisson process $Y_i \sim \pi(\lambda_i)$, and $\lambda_i = \lambda + \lambda \sum_{j=1}^{q_i} p_j^m$, in which, p_j^m denotes the cache miss probability of node v_j , and q_i indicates the number of neighbor nodes of node v_i .

Proof: In a CCMAN, the consumer node located at the edge of the network won't receive interest packet from others. Hence, their content request arrival process follow $X \sim \pi(\lambda)$, which is the interest packet distribution of their nodes as well.

According to the property of Poisson distribution, the Poisson process with rate λ filtered by the cache miss probability of p_i^m still follows the Poisson distribution with parameter λp_i^m . Therefore, the missed content request arrival process at each node follows Poisson process.

Since the sum of two independent Poisson process is still a Poisson process, we can conclude that the content request arrival event at node v_i , $Y_i \sim \pi(\lambda_i)$ and

$$\lambda_i = \lambda + \lambda \sum_{j=1}^{q_i} p_j^m, \quad (1)$$

where q_i indicates the number of neighbor nodes of the node v_i . Obviously, in a line topology, $q_i = 1$ and in a binary tree topology, $q_i = 2$. ■

Lemma 2. The relationship of cache frequency g_i and cache probability p_i^c of node v_i can be expressed by

$$g_i = p_i^c R [1 - \exp(-(\lambda_i t)/R)]/t. \quad (2)$$

Proof: Assuming there are R contents served by content server and $B^r(0, t)$ denotes whether there is request for content r during the period $(0, t)$. If there is at least one

request, $B^r(0, t) = 1$; otherwise, $B^r(0, t) = 0$. $S_i(0, t)$ indicates the number of all different requests received by node v_i during the period $(0, t)$. So $S_i(0, t)$ is calculated as

$$S_i(0, t) = \sum_{r=1}^R B^r(0, t). \quad (3)$$

The mathematical expectation of the sum of the different request received by v_i can be expressed as

$$\begin{aligned} \mathbf{E}[S_i(0, t)] &= \mathbf{E}\left[\sum_{r=1}^R B^r(0, t)\right] \\ &= \sum_{r=1}^R P(B^r(0, t) \geq 1) \\ &= \sum_{r=1}^R [1 - P(B^r(0, t) = 0)] \\ &\stackrel{(a)}{=} R[1 - \exp(-(\lambda_i t)/R)]. \end{aligned} \quad (4)$$

Due to $B^r(0, t) = 0$ denotes that there is no content request arrives during $(0, t)$. Therefore, (a) employs the Poisson process and $P(B^r(0, t) = 0) = \exp(-(\lambda_i t)/R)$.

According to the definition of cache frequency of node v_i , g_i is calculated as

$$\begin{aligned} g_i &= p_i^c E[S_{m, c_i}(0, t)]/t \\ &= p_i^c R [1 - \exp(-(\lambda_i t)/R)]/t. \end{aligned} \quad (5)$$

■

When the system reaches steady state, in other words, when $R \rightarrow \infty$, $t \rightarrow \infty$, g_i is calculated as

$$\begin{aligned} &\lim_{t \rightarrow \infty} \lim_{R \rightarrow \infty} g_i \\ &= \lim_{t \rightarrow \infty} \lim_{R \rightarrow \infty} \{p_i^c R [1 - \exp(-(\lambda_i t)/R)]/t\} \\ &= \lim_{t \rightarrow \infty} \lim_{R \rightarrow \infty} \{-p_i^c R (-\lambda_i t/R)/t\} \\ &= p_i^c \lambda_i. \end{aligned} \quad (6)$$

It means when the number of contents in the network is infinite, the probability of having the same content request is zero, and the node will cache every passing by content in a probability p_i^c .

Lemma 3. The cache hit probability of the node v_i in the LRU satisfies

$$p_i^h = (1 - p_i^r) \left\{ p_i^c \sum_{j=1}^{k-2} [(1 - p_i^c)(1 - p_i^r)]^j + p_i^c \right\}, \quad (7)$$

in which, the content replacement probability of node v_i , p_i^r is

$$p_i^r = \exp\left(-\frac{\lambda_i x_i t}{p_i^c R^2 (\exp(-\lambda_i t/R))}\right), \quad (8)$$

where x_i is the cache space size of node v_i .

Proof: In the LRU, the hit of the content r at node v_i must satisfy the following two conditions simultaneously: 1) the content r locates at the top of the CS after its last request; 2) the data size of cached chunks at v_i is not more than the cache space size x_i after the condition 1) occurs.

First of all, we deduce the content replacement probability of the node v_i under condition 1) as

$$p_i^r = P[p_i^c S_i(0, \tau^r) > x_i] = P\left[\tau^r > \frac{x_i}{g_i}\right] \\ = \exp\left(-\frac{\lambda_i}{R g_i} x_i\right) = \exp\left(-\frac{\lambda_i x_i t}{p_i^c R^2 (\exp(-\lambda_i t/R))}\right). \quad (9)$$

From (8), we can know that, when total content number R is set to a certain specified value, the miss probability is a function of the node cache space x_i , content request arrival rate λ_i and cache probability p_i^c .

Obviously, the condition 1) consists two situations with LRU: a) miss of content r occurs and node v_i cached it on the last request; b) hit of content r occurs. The state whether content r located at the top of CS at the time of the k^{th} content request arrived at node v_i is related with the its previous state. Suppose that $p_i^{t,k}$ denotes the probability of that the content r is located at the top of CS while the k^{th} request for it. When $k = 1$, apparently, $p_i^{t,1} = p_i^c$, and when $k = 2$,

$$p_i^{t,2} = p_i^{t,1} (p_i^r p_i^c + (1 - p_i^r)) + (1 - p_i^{t,1}) p_i^c \\ = p_i^c (1 - p_i^r) (1 - p_i^c) + p_i^c. \quad (10)$$

Assuming that $p_i^p = (1 - p_i^r) (1 - p_i^c)$, then when $k = 3$, similarly, $p_i^{t,3} = p_i^{t,2} p_i^p + (1 - p_i^{t,2}) p_i^c = p_i^c p_i^p + p_i^c p_i^p + p_i^c$. Hence, the probability of that the content r is located at the top of CS at the time of the k^{th} ($k \geq 2$) content request arrived node is

$$p_i^{t,k} = p_i^c \sum_{j=1}^{k-1} p_i^{p,j} + p_i^c. \quad (11)$$

Based on the above analysis, the cache hit probability of k^{th} request at node v_i is, $p_i^h = p_i^{t,k-1} (1 - p_i^r)$. Then, we have the whole cache hit probability at node v_i

$$p_i^h = (1 - p_i^r) \left\{ p_i^c \sum_{j=1}^{k-2} [(1 - p_i^c) (1 - p_i^r)]^j + p_i^c \right\}. \quad (12)$$

Assume that in the CCMAN, the number of nodes is N . From **Lemma 1** and **Lemma 3**, the cache hit ratio of the whole network can be calculated as follows

$$h = \frac{\sum_{i=1}^N \lambda_i p_i^h}{\sum_{i=1}^N \lambda_i}. \quad (13)$$

The cache space size of node v_i is x_i , and from (13), we can obtain the cache utility η as

$$\eta = \frac{h}{\sum_{i=1}^N x_i} = \frac{\sum_{i=1}^N \lambda_i p_i^h}{\left(\sum_{i=1}^N \lambda_i\right) \left(\sum_{i=1}^N x_i\right)}. \quad (14)$$

From (14), we can see that, cache utility is decided by the cache hit probability, the content request arrival rate and the occupied cache space.

In a specific network topology, given the content request generating rate λ , the cache utility is decided by the cache hit probability and the cache size of each node. So we will design a caching placement strategy to reach the maximal cache utility for best tradeoff between cache hit ratio performance and the occupied cache space. In doing so, we select some "important" nodes in a given network topology of CCMAN, which can provide content retrieval delay guarantee and reduce content replicas redundancy.

V. CSEC SCHEME AND PERFORMANCE ANALYSIS

A. Proposed CSEC Scheme based on GDS

For a given undirected connected graph $G = \langle V, E \rangle$, with V and E denoting the point set and link set of G respectively, the definition of DS and GDS are given as following.

Definition 7. Suppose a point set $S \subset V$, $\forall v \in V - S$, $\exists u \in S$, such that (u, v) is a subset of E , which denotes the link between the node u and node v , then S is denoted as the DS of G .

Definition 8. For a given parameter m , suppose a point set $\hat{S} \subset V$, $\forall v \in V - \hat{S}$, $\exists u \in \hat{S}$, such that $v \in \hat{N}_m(u)$, which denotes the set of nodes whose shortest distance to node u is not more than m hops, then \hat{S} is defined as the m-GDS of G .

The selection of GDS in a undirected connected graph is given in Algorithm 1.

Algorithm 1 m-GDS selection

Initialization:

- 1) In a given undirected connected graph $G = \langle V, E \rangle$, predefine a parameter m ;
- 2) Mark all nodes V in white and compute the degree of each node in G ;
- 3) Denote $\hat{N}_m(u)$ as the set of nodes whose shortest path to node u is not more than m ;
- 4) Denote $\hat{d}(v)$ as the number of white node in $\hat{N}_m(v)$.

Step 1:

- 1) Select the node v with the largest degree and mark it in black;
- 2) mark the white nodes in $\hat{N}_m(v)$ in gray.

Step 2:

- 1) Compute $\hat{d}(v)$ of the gray node v ;
- 2) Mark the node v with the largest $\hat{d}(v)$ in black;
- 3) Mark the white node in $\hat{N}_m(v)$ in gray;
- 4) Repeat Step 2 until all the nodes in v are marked in black or gray.

Output:

- 1) Dominating nodes: the nodes are marked in black, denoted by c_1, c_2, \dots, c_n ;
 - 2) Normal nodes: the nodes are marked in gray.
-

According to the definition of GDS, each dominating node covers several normal nodes. Since the graph G is connected, the algorithm can be converged after all the nodes are marked.

In fact, the value of m can be seen as the the criterion for finding an m-GDS, which controls the size of the dominate set. Based on the definition of m-GDS, the nodes in $V - \hat{S}$ must be within m hops away from at least one node in \hat{S} . m-GDS will be equivalent to DS when $m = 1$.

Lemma 4. Let L denote the maximum hops of the shortest path between the content server and other nodes in the network. \hat{S} is an empty set when $m \geq L$, therefore, m is in the range of $[1, L-1]$.

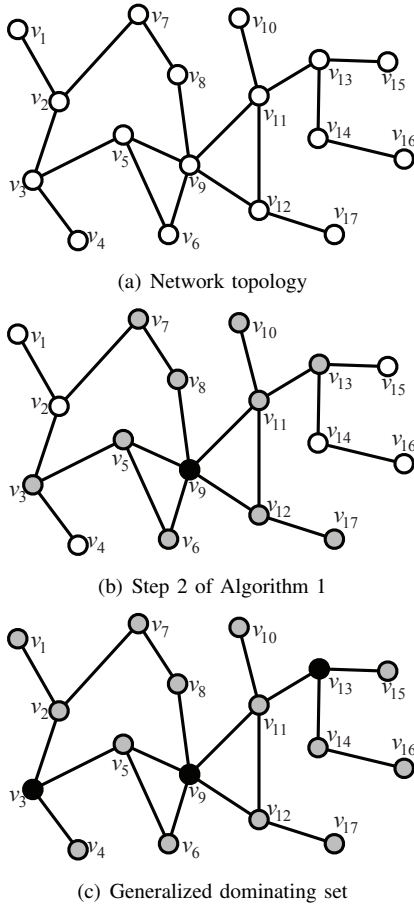


Fig 4. Example of GDS selection ($m=2$)

Taking $m = 2$ as an example, the schematic diagram of GDS selection algorithm is depicted in Fig. 4. Fig. 4-a) describes the network topology, while Fig. 4-b) shows the Step 2 of **Algorithm 1**, where node v_9 with the largest degree is marked in black and its neighbor nodes in 2 hops are marked in gray. The final state is shown in Fig. 4-c) where node v_3 , v_9 and v_{13} are selected as the generalized dominating nodes.

Assuming that the radio coverage of each wireless node is the same, an arbitrary topology of CCMAN can be represented by an undirected connected graph $G = \langle V, E \rangle$. An element v of set V denotes a user in CCMAN, collectively called node hereinafter. An element (u, v) of set E indicates that node u and node v are located within the radio coverage of each other, in other words, node u and node v can contact with each other directly. Therefore, the method of m-GDS can be used to achieve the coverage region partitioning for CCMAN.

In this paper, we propose a CSEC scheme based on the m-GDS in CCMAN, in which, the content replicas are saved in the nodes in the set of \hat{S} . For a node v_i in \hat{S} , the caching probability is one, otherwise, the caching probability of the node v_i is set to be zero. The detailed process of the proposed CSEC is given in **Algorithm 2**. by the definition of m-GDS it can guarantee that the contents can be retrieved within m hops.

The computational complexity of CSEC is mainly depend on that of GDS. It has been proved that the computational complexity of GDS obtained by **Algorithm 1** is $O(|E||G|)$ [44].

Noted that when employing CSEC, the network controller functionality of CCMAN can be implemented by SDN. When the network topology varies due to wireless node mobility, the network controller will update the proposed scheme procedure.

Algorithm 2 CSEC scheme

Initialization:

The network controller decides the value of m .

Caching Placement:

For each content, the network controller operates as following steps:

Step 1:

The network controller obtains the GDS \hat{S} taking the content server as the starting node according to **Algorithm 1**.

Step 2:

The network controller configures the caching probability p_i^c of node v_i according to

$$p_i^c = \begin{cases} 1, & v_i \in \hat{S} \\ 0, & v_i \notin \hat{S} \end{cases} \quad (15)$$

Step 3:

If the caching probability p_i^c of node v_i is one, when node v_i receives a Data packet matched in the PIT, it forwards the Data packet and caches a content duplicate;

If the caching probability p_i^c of node v_i is zero, when node v_i receives a Data packet matched in the PIT, it forwards the Data packet only.

B. Performance Analysis of CSEC

In this subsection, we study the cache hit probability characterization of each caching node in the proposed CSEC scheme. Based on this, we use the cache utility to evaluate the network performance of CSEC analytically.

The analytical model proposed in this paper is extent from the IRM [38]. Compared with the study on the line topology of [38], two main differences of the arbitrary topology are taken into consideration in this paper, 1) the nodes act as routers as well as users; 2) the content request at a node may come from any neighboring nodes. Once there is a cache miss at a node, the content request will be forwarded to its next hop (neighboring node). In this model, the caches are managed by LRU caching replacement policy.

We assume that, there are N nodes in the link P , which can be represented as $P = v_1 \rightarrow v_2 \rightarrow \dots \rightarrow v_{N-1} \rightarrow v_N$, and v_1 is the farthest node from the content server, v_N is the content server. The maximum hops of the shortest path between the content server and other nodes in the network is denoted as L . In the link P , $L = N - 1$. From **Lemma 4** we can deduce that, the parameter m of m-GDS is in the range of $[1, L - 1]$. When the m is given, the number of the caching nodes is

$$n_m = \left\lceil \frac{N-1}{m} \right\rceil - 1. \quad (16)$$

These caching nodes are indicated as c_1, c_2, \dots, c_{n_m} . We assume that the nodes covered by node c_i and c_{i+1} are serviced by c_{i+1} .

Proposition 1: The cache hit probability of the caching node c_i in the GDSC placement strategy and the LRU re-

placement strategy is

$$p_{c_i}^h = 1 - \exp \left(- \frac{xt\lambda_{c_i}}{R^2 \left[1 - \exp \left(- \frac{\lambda_{c_i}t}{R} \right) \right]} \right), \quad (17)$$

where λ_{c_i} is the content request arrival rate at the caching node c_i .

Proof: First, we deduce the content request arrival rate at the caching node c_i .

Obviously, the content request arrival rate at the node v_1 is $\lambda_1 = \lambda$, i. e., the content request generates from itself. The content request at the node v_2 consists of two parts, content request generating from itself and missed content request from its neighbor nodes. Assume the degree of the node v_i is d_i . The content request arrival rate at the node v_2 is $\lambda_2 = \lambda + \lambda(d_2 - 1)$.

By parity of reasoning, the content request arrival rate at the node v_i , which is in the region of caching node c_1 , is

$$\lambda_i = \begin{cases} \lambda, & i = 1, \\ \lambda \left(\sum_{j=2}^i (d_j - 1) + 1 \right), & 2 \leq i \leq y, \end{cases} \quad (18)$$

where $y = N - n_m m$ and d_j denotes the degree of the node v_j . So, the content request arrival rate at the caching node c_1 is $\lambda_{c_1} = \lambda_y = \lambda \sum_{j=2}^y (d_j - 1)$.

According to (15), the cache probability of caching node c_1 is $p_{c_1}^c = 1$, then from (8), the content replacement probability of caching node c_1 is

$$p_{c_1}^r = \exp \left(- \frac{\lambda_{c_1} xt}{R^2 (\exp(-\lambda_{c_1} t / R))} \right). \quad (19)$$

The content requests arrived at the caching node c_2 come from the content requests of the nodes in the cover region of c_2 and missed content requests from the caching node c_1 . So, the content request arrival rate at the caching node c_2 is $\lambda_{c_2} = \lambda_{c_1} p_{c_1}^m + \lambda \sum_{j=y+1}^{y+m} (d_j - 1)$.

By the same way, the content requests arrived at the caching node c_i can be expressed as

$$\lambda_{c_i} = \lambda_{c_{i-1}} p_{c_{i-1}}^m + \lambda \sum_{j=y'(i)+1}^{y'(i)+m} (d_j - 1), \quad (20)$$

where $y'(i) = y + (i - 2)m$.

Then, the content replacement probability at the caching node c_i can be obtained from (8), which is

$$p_{c_i}^r = \exp \left(- \frac{xt\lambda_{c_i}}{R^2 \left[1 - \exp \left(- \frac{\lambda_{c_i}t}{R} \right) \right]} \right). \quad (21)$$

It is worth noting that, when a content is replaced at the caching node c_i , there will be a miss of the correspond content request, so the content replacement probability at the caching

node c_i is equivalent to the caching miss probability of the caching node c_i . We have,

$$p_{c_i}^m = \exp \left(- \frac{xt\lambda_{c_i}}{R^2 \left[1 - \exp \left(- \frac{\lambda_{c_i}t}{R} \right) \right]} \right). \quad (22)$$

From (22), the cache hit probability of caching node c_i can be obtained correspondingly,

$$p_{c_i}^h = 1 - p_{c_i}^m = 1 - \exp \left(- \frac{xt\lambda_{c_i}}{R^2 \left[1 - \exp \left(- \frac{\lambda_{c_i}t}{R} \right) \right]} \right), \quad (23)$$

where, the content request arrival rate at the caching node c_i is given in (20). ■

Finally, according to (20) and (23), the cache hit ratio of the entire network is calculated as

$$h = \frac{\sum_{i=1}^{n_m} \lambda_{c_i} p_{c_i}^h}{\sum_{j=1}^N \lambda_j}. \quad (24)$$

Then the cache utility is

$$\eta = \frac{\sum_{i=1}^{n_m} \lambda_{c_i} p_{c_i}^h}{\left(\sum_{j=1}^N \lambda_j \right) \left(\sum_{i=1}^{n_m} x_i \right)}. \quad (25)$$

VI. PERFORMANCE EVALUATION

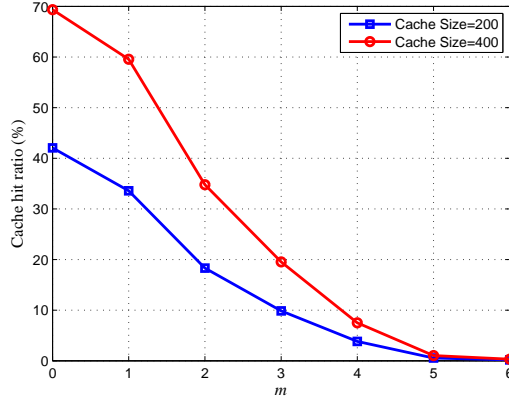
Performance of the proposed CSEC scheme is evaluated by the ndnSIM [45] simulator. The improved random network topology generation [46] is employed in this paper. Considering the application scenario of CCMAN, some of the wireless nodes provide the contents for others nodes as content servers (also termed as content producers), and all the nodes can request the contents as content consumers. The best route strategy is used for content forwarding, and LRU is used for caching replacement strategy. We assume that each content has equal size, and the cache size of each node is measured by the number of contents. The parameter settings of the simulation are given in Table II.

TABLE II: Parameters of Simulation

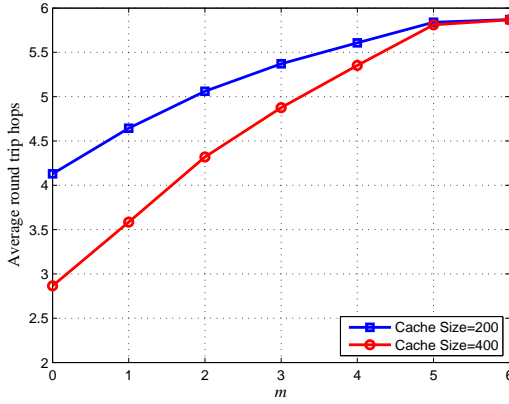
Item	Setting
Network topology	Random network
Content request model	Poisson process with $\lambda = 2$
Content requests of each node	1000
Number of wireless nodes	100
Number of consumers	100
Number of producers	4

Firstly, the cache hit ratio, ARTH and ARTT are selected as the criterion to compare the network performance of the proposed CSEC and LCE. The network performance with varying m hops is given in Fig. 5.

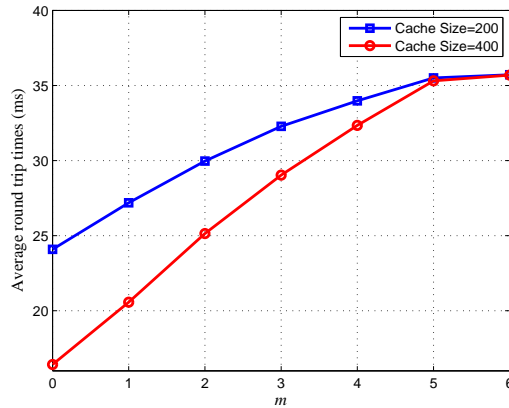
From Fig. 5 we can see that the cache hit ratio, ARTH and ARTT performance deteriorate with the increasing m . Obviously, the LCE, shown in the simulation results of $m = 0$, has the best network performance. In a given network topology, the



(a) Cache hit ratio



(b) ARTH

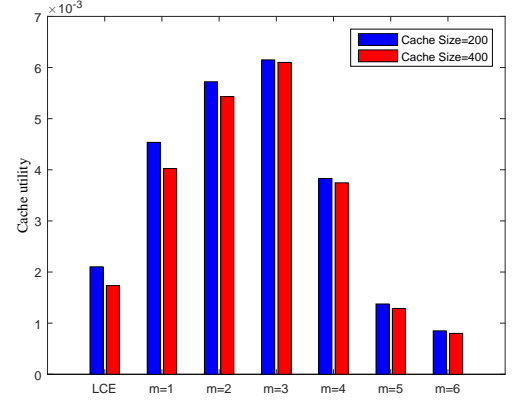


(c) ARTT

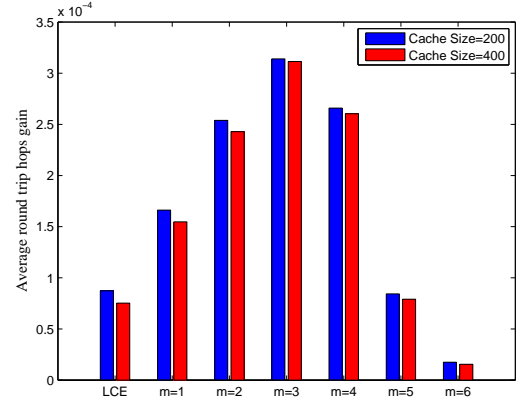
Fig 5. Network performance comparison with varying m .

smaller value of m means that the number of nodes involved content caching is larger. From this point of view, the LCE is the worst case, the contents should be cached in all the nodes in the network to fulfill the condition of $m=0$. This will cost all the cache space of all the wireless nodes in CCMAN, which should be avoided because of the limitation of the cache size in CCMAN. So the performance gain in term of the involved caching nodes numbers, i.e., totally occupied cache space in CCMAN, should be taken fully consideration in the caching strategy design.

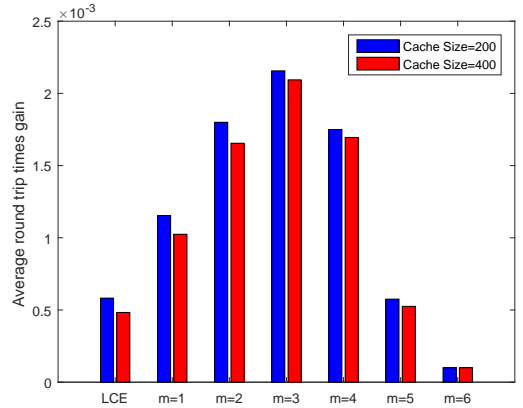
The performance gains with varying m are evaluated in



(a) Cache utility



(b) ARTH gain



(c) ARTT gain

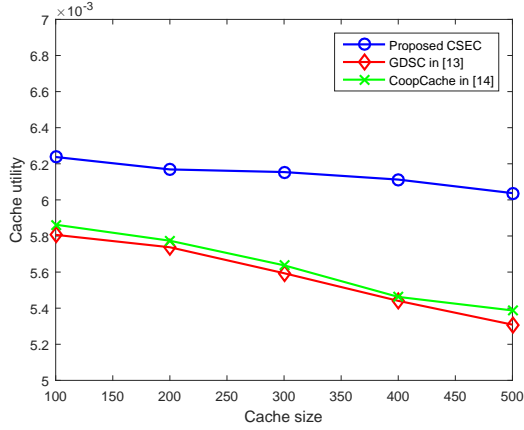
Fig 6. Performance gain comparison with varying m .

Fig. 6. The performance gains includes the cache utility, ARTH gain and ARTT gain. The cache utility is defined in **Definition 6**. The ARTH gain is defined as

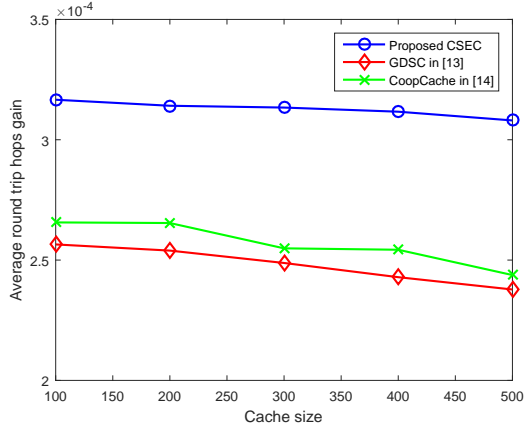
$$ARTH \text{ gain} = \frac{ARTH \text{ of } LCE - ARTH \text{ of } CSEC}{Occupied \text{ cache space}} \quad (26)$$

The ARTT gain is defined as the same way of ARTH gain.

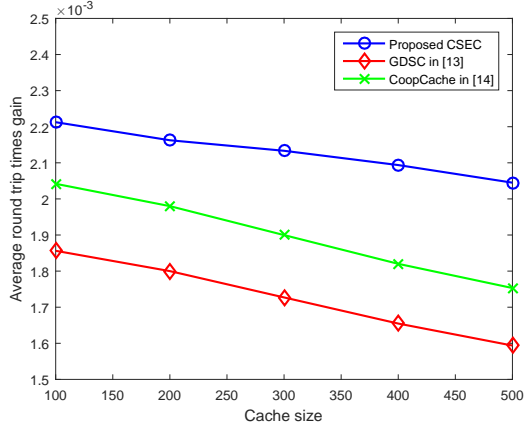
The cache utility varying with m are given in Fig. 6-(a). We should note that in the simulation $m=0$ stands for the LCE scheme. We can see that, the largest value of cache utility can



(a) Cache utility



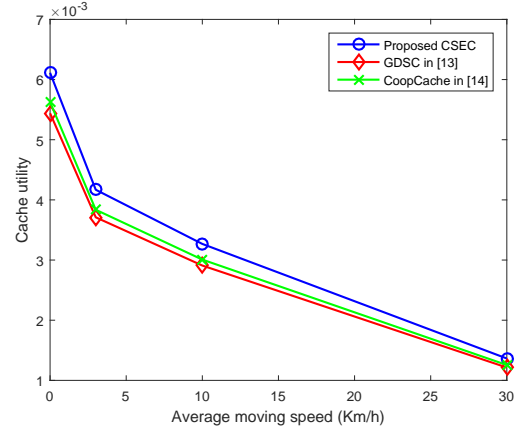
(b) ARTH gain



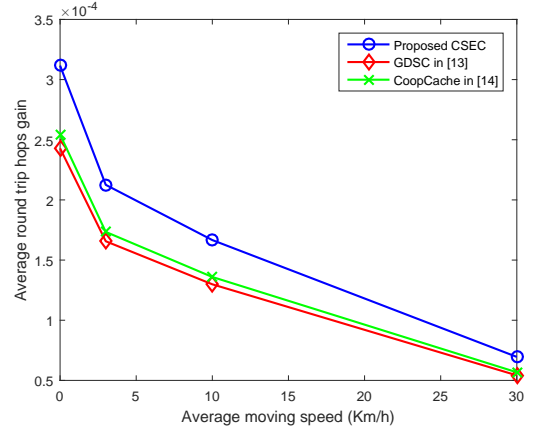
(c) ARTT gain

Fig 7. Performance gain comparison with varying cache size.

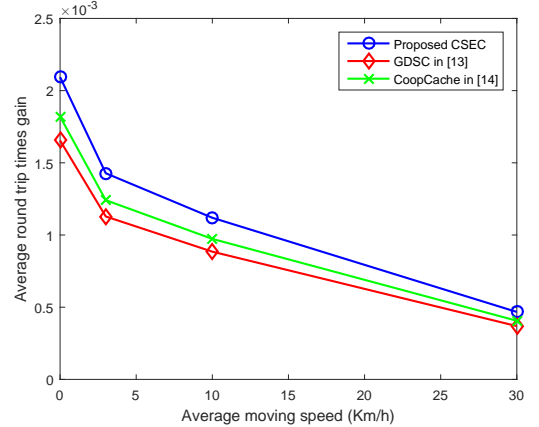
be obtained when $m = 3$. The ARTH gain and ARTT gain varying with m are given in Fig. 6-(b), Fig. 6-(c). We can see that, the largest value of ARTH gain and ARTT gain can be obtained when $m = 3$. The simulation results indicate that, in a given CCMAN, the optimal cache utility, ARTH gain and ARTT gain can be obtained by the proposed CSEC. Besides, the simulation results indicate the impact of the cache space of wireless node on the network performance and cache gain. In Fig. 6-(a), the cache utility value are nearly the same even though the cache size is different. It means that the cache size of wireless node has less impact on the cache utility. However,



(a) Cache utility



(b) ARTH gain



(c) ARTT gain

Fig 8. Performance gain comparison with varying moving speed.

we should note that the larger node cache size can obtain better performance on the cache hit ratio, ARTH and ARTT as shown in Fig. 5.

Furthermore, the caching performance of the proposed CSEC is compared with the existing DS based caching strategies, GDSC [12] and CoopCache [13]. The GDSC is a caching placement scheme aimed at network power consumption minimization. The CoopCache is extended from GDSC and considers the local content popularity for content replacement in the cache store.

The caching performance comparisons in the stable wireless

nodes are shown in Fig. 7. We can see that, CSEC proposed in this paper outperforms CoopCache and GDSC on cache utility, ARTH gain and ARTT gain. The reason is that, the proposed CSEC is dedicated to improve the cache space utilization while GDSC selects the caching nodes to minimize the network power consumption and CoopCache makes the cache replacement decision to improve the cache hit ratio. As a result, both the GDSC and CoopCache cost more cache space than CSEC and perform worse than CSEC on cache utility, ARTH gain and ARTT gain.

The impact of wireless node mobility on the caching performance is verified by simulation. The caching performance gain comparisons with various node moving speed are given in Fig. 8. The simulation results show that the cache utility, ARTH gain and ARTT gain decrease when the moving speed increases, since the mobility of wireless nodes makes the wireless channel states and the data transmission rate variable, which will impact the caching performance. In the mobility scenario, the proposed CSEC obtains better caching performance gains than that of GDSC and CoopCache. We notice that the gains of CSEC decrease compared to GDSC and CoopCache when the moving speed is increasing, since the data rate of wireless links are deteriorated severely when the moving speed of nodes is high.

VII. CONCLUSION

In this paper, we studied the cache space efficient in-networking caching in CCMAN. The theoretical performance of caching was studied, especially, the cache utility expression for tradeoff between caching hit ratio and the occupied cache space is deduced. The theoretical performance analysis on the cache utility provided a basis for the caching placement scheme design. In order to reach the maximal cache utility, a CSEC scheme was proposed. The analytical performance of CSEC in terms of the cache hit ratio and cache utility was derived. Simulation results showed that the proposed CSEC obtained performance improvement on cache utility, ARTH gain and ARTT gain which means the efficient cache space utilization can be achieved when the network topology of CCMAN is given. Meanwhile, the proposed CSEC obtains the best caching performance compared with the traditional caching schemes in the scenarios of both the stable and mobile networks.

REFERENCES

- [1] J. Kurose, "Information-centric networking: The evolution from circuits to packets to content," *Computer Networks*, vol. 66, pp. 112–120, 2014.
- [2] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard, "Networking named content," in *Proceedings of the 5th international conference on Emerging networking experiments and technologies*. ACM, 2009, pp. 1–12.
- [3] J. Loo and M. Aiash, "Special issue on information-centric network architecture, protocols, algorithms and applications," *Journal of Network and Computer Applications*, vol. 50, pp. 60–63, 2014.
- [4] M. Amadeo, C. Campolo, and A. Molinaro, "Forwarding strategies in named data wireless ad hoc networks: Design and evaluation," *Journal of Network and Computer Applications*, vol. 50, pp. 148–158, 2015.
- [5] M. Amadeo, C. Campolo, A. Molinaro, and G. Ruggeri, "Content-centric wireless networking: A survey," *Computer Networks*, vol. 72, pp. 1–13, 2014.
- [6] M. Varvello, I. Rimac, U. Lee, L. Greenwald, and V. Hilt, "On the design of content-centric MANETs," in *2011 International Conference on Wireless On-Demand Network Systems and Services*, Jan. 2011, pp. 1–8.
- [7] M. Amadeo, C. Campolo, and A. Molinaro, "Design and analysis of a transport-level solution for content-centric VANETs," in *2013 IEEE International Conference on Communications Workshops (ICC)*, Jun. 2013, pp. 532–537.
- [8] E. Talipov, Y. Chon, and H. Cha, "Content sharing over smartphone-based delay-tolerant networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 3, pp. 581–595, Mar. 2013.
- [9] W. S. Jung, H. Ahn, and Y. B. Ko, "Designing content-centric multi-hop networking over Wi-Fi direct on smartphones," in *2014 IEEE Wireless Communications and Networking Conference (WCNC)*, Apr. 2014, pp. 2934–2939.
- [10] G. Zhang, Y. Li, and T. Lin, "Caching in information centric networking: A survey," *Computer Networks*, vol. 57, no. 16, pp. 3128–3141, 2013.
- [11] Y. Xu, Y. Li, T. Lin, G. Zhang, Z. Wang, and S. Ci, "A dominating-set-based collaborative caching with request routing in content centric networking," in *2013 IEEE International Conference on Communications (ICC)*, Jun. 2013, pp. 3624–3628.
- [12] L. Zhou, T. Zhang, X. Xu, Z. Zeng, and Y. Liu, "Generalized dominating set based cooperative caching for content centric ad hoc networks," in *2015 IEEE/CIC International Conference on Communications in China (ICCC)*, Nov. 2015, pp. 1–6.
- [13] Y. Liu, D. Zhu, and W. Ma, "A novel cooperative caching scheme for content centric mobile ad hoc networks," in *2016 IEEE Symposium on Computers and Communication (ISCC)*, Jun. 2016, pp. 824–829.
- [14] N. Laoutaris, S. Syntila, and I. Stavrakakis, "Meta algorithms for hierarchical Web caches," in *IEEE International Conference on Performance, Computing, and Communications*, 2004, 2004, pp. 445–452.
- [15] Y. Xu, Y. Li, T. Lin, Z. Wang, W. Niu, H. Tang, and S. Ci, "A novel cache size optimization scheme based on manifold learning in content centric networking," *Journal of Network and Computer Applications*, vol. 37, pp. 273–281, 2014.
- [16] D. Rossi and G. Rossini, "On sizing CCN content stores by exploiting topological information," in *2012 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Mar. 2012, pp. 280–285.
- [17] C. Bernardini, T. Silverston, and O. Festor, "Socially-aware caching strategy for content centric networking," in *2014 IFIP Networking Conference*, Jun. 2014, pp. 1–9.
- [18] L. Wu, T. Zhang, X. Xu, Z. Zeng, and Y. Liu, "Grey relational analysis based cross-layer caching for content centric networking," in *2015 IEEE/CIC International Conference on Communications in China (ICCC)*, Nov. 2015, pp. 1–6.
- [19] S. H. Lim, Y. B. Ko, G. H. Jung, J. Kim, and M. W. Jang, "Inter-chunk popularity-based edge-first caching in content-centric networking," *IEEE Communications Letters*, vol. 18, no. 8, pp. 1331–1334, Aug. 2014.
- [20] W. K. Chai, D. He, I. Psaras, and G. Pavlou, "Cache less for more in information-centric networks (extended version)," *Computer Communications*, vol. 36, no. 7, pp. 758–770, 2013.
- [21] K. Cho, M. Lee, K. Park, T. T. Kwon, Y. Choi, and S. Pack, "Wave: Popularity-based and collaborative in-network caching for content-oriented networks," in *2012 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Mar. 2012, pp. 316–321.
- [22] H. Salah, B. Schiller, and T. Strufe, "CoMon: A system architecture for improving caching in CCN," in *2014 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Apr. 2014, pp. 213–214.
- [23] Y. Wang, M. Xu, and Z. Feng, "Hop-based probabilistic caching for information-centric networks," in *2013 IEEE Global Communications Conference (GLOBECOM)*, Dec. 2013, pp. 2102–2107.
- [24] I. Psaras, W. K. Chai, and G. Pavlou, "Probabilistic in-network caching for information-centric networks," in *Proceedings of the second edition of the ICN workshop on Information-centric networking*. ACM, 2012, pp. 55–60.
- [25] G. Rossini and D. Rossi, "Evaluating CCN multi-path interest forwarding strategies," *Computer Communications*, vol. 36, no. 7, pp. 771–778, 2013.
- [26] Y. Li, T. Zhang, X. Xu, Z. Zeng, and Y. Liu, "Content popularity and node level matched based probability caching for content centric networks," in *2016 IEEE/CIC International Conference on Communications in China (ICCC)*, Jul. 2016, pp. 1–6.
- [27] G. Cao, L. Yin, and C. R. Das, "Cooperative cache-based data access in ad hoc networks," *Computer*, vol. 37, no. 2, pp. 32–39, Feb. 2004.

- [28] C. y. Chow, H. V. Leong, and A. T. s. Chan, "Grococa: group-based peer-to-peer cooperative caching in mobile environment," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 1, pp. 179–191, Jan. 2007.
- [29] L. Yin and G. Cao, "Supporting cooperative caching in ad hoc networks," *IEEE Transactions on Mobile Computing*, vol. 5, no. 1, pp. 77–89, Jan. 2006.
- [30] M. Fiore, C. Casetti, and C. F. Chiasserini, "Caching strategies based on information density estimation in wireless ad hoc networks," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 5, pp. 2194–2208, Jun. 2011.
- [31] Y. Ma and A. Jamalipour, "A cooperative cache-based content delivery framework for intermittently connected mobile ad hoc networks," *IEEE Transactions on Wireless Communications*, vol. 9, no. 1, pp. 366–373, Jan. 2010.
- [32] S. B. Lee, S. H. Y. Wong, K. W. Lee, and S. Lu, "Content management in a mobile ad hoc network: Beyond opportunistic strategy," in *2011 Proceedings IEEE INFOCOM*, Apr. 2011, pp. 266–270.
- [33] L. Zhou, T. Zhang, X. Xu, Z. Zeng, and Y. Liu, "Broadcasting based neighborhood cooperative caching for content centric ad hoc networks," in *2015 IEEE/CIC International Conference on Communications in China (ICCC)*, Nov. 2015, pp. 1–5.
- [34] W. Quan, C. Xu, J. Guan, H. Zhang, and L. A. Grieco, "Social cooperation for information-centric multimedia streaming in highway VANETs," in *Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks 2014*, Jun. 2014, pp. 1–6.
- [35] L. Wang, A. Afanasyev, R. Kuntz, R. Vuyyuru, R. Wakikawa, and L. Zhang, "Rapid traffic information dissemination using named data," in *Workshop on emerging name-oriented mobile networking design-architecture, algorithms, and applications*. ACM, 2012, pp. 7–12.
- [36] T. Zhang, H. Fan, J. Loo, and D. Liu, "User preference aware caching deployment for device-to-device caching networks," *IEEE Systems Journal*, vol. PP, no. 99, pp. 1–12, 2017.
- [37] E. J. Rosensweig, J. Kurose, and D. Towsley, "Approximate models for general cache networks," in *2010 Proceedings IEEE INFOCOM*, Mar. 2010, pp. 1–9.
- [38] G. Carofiglio, M. Gallo, and L. Muscariello, "On the performance of bandwidth and storage sharing in information-centric networks," *Computer Networks*, vol. 57, no. 17, pp. 3743–3758, 2013.
- [39] Y. Zhu, Z. Mi, and W. Wang, "A cache probability replacement policy based on content popularity in content centric networks," *J. Electronics and Information Technology*, vol. 35, no. 6, pp. 1305–1310, 2013.
- [40] E. Leonardi and G. Luca, "Least recently used caches under the shot noise model," in *2015 IEEE Conference on Computer Communications (INFOCOM)*, Apr. 2015, pp. 2281–2289.
- [41] M. Leconte, G. Paschos, L. Gkatzikis, M. Draief, S. Vassilaras, and S. Chouvardas, "Placing dynamic content in caches with small population," in *2016 IEEE Conference on Computer Communications (INFOCOM)*, Apr. 2016, pp. 1–9.
- [42] V. Ramaswamy, "Modeling and performance analysis of information-centric networks," in *2016 25th International Conference on Computer Communication and Networks (ICCCN)*, Aug 2016, pp. 1–8.
- [43] R. NAKAMURA and H. OHSAKI, "Performance analysis of content-centric networking on an arbitrary network topology," *IEICE Transactions on Communications*, vol. advpub, 2017.
- [44] S. Guha and S. Khuller, "Approximation algorithms for connected dominating sets," *Algorithmica*, vol. 20, no. 4, pp. 374–387, Apr 1998. [Online]. Available: <https://doi.org/10.1007/PL00009201>
- [45] S. Mastorakis, A. Afanasyev, I. Moiseenko, and L. Zhang, "ndnSIM 2.0: A new version of the NDN simulator for NS-3," *NDN, Technical Report NDN-0028*, 2015.
- [46] T. Zhang, J. Cao, Y. Chen, L. Cuthbert, and M. Elkashlan, "A small world network model for energy efficient wireless networks," *IEEE Communications Letters*, vol. 17, no. 10, pp. 1928–1931, Oct. 2013.



Tiankui Zhang (M'10-SM'15) received the Ph.D. degree in Information and Communication Engineering and B.S. degree in Communication Engineering from Beijing University of Posts and Telecommunications (BUPT), China, in 2008 and 2003, respectively. Currently, he is an Associate Professor in School of Information and Communication Engineering at BUPT. His research interests include wireless communication networks, green wireless networks, signal processing for wireless communications, content centric wireless networks. He had published more than 100 papers including journal papers on *IEEE Journal on Selected Areas in Communications*, *IEEE Transaction on Communications*, etc., and conference papers, such as *IEEE GLOBECOM* and *IEEE ICC*.



Xiaogeng Xu received her B.S. degree from Beijing University of Technology, China, in 2005, and M.S. degree from Chalmers University of Technology (CTH), Sweden, in 2007. She is currently pursuing her Ph.D. degree in Information and Communication Engineering from Beijing University of Posts and Telecommunications. Her current research interests include caching performance analysis and mechanism design in content centric networks.



Le Zhou received B.S. degree in Electronic and Communication Engineering and M.S. in Information and Communication Engineering from Beijing University of Posts and Telecommunications, China, in 2013 and 2016, respectively. Her research interests include caching performance analysis and mechanism design in content centric networks.



Xinwei Jiang received his B.S. degree from Nanjing University of Posts and Telecommunications, China, in 2015. He is currently pursuing his M.S. degree in Information and Communication Engineering from Beijing University of Posts and Telecommunications. His current research interests include routing and caching strategy in content centric networks.



Jonathan Loo received his M.Sc. degree in Electronics (with Distinction) and the Ph.D. degree in Electronics and Communications from the University of Hertfordshire, Hertfordshire, U.K., in 1998 and 2003, respectively. Between 2003 and 2010, he was a Lecturer in Multimedia Communications with the School of Engineering and Design, Brunel University, Uxbridge, U.K. Between June 2010 and May 2017, he was an Associate Professor in Communication Networks at the School of Science and Technology, Middlesex University, London, U.K.

From June 2017, he is a Chair Professor in Computing and Communication Engineering at the School of Computing and Engineering, University of West London, United Kingdom. His research interests include information centric networking, cloud computing, wireless/mobile networks and protocols, network security, wireless communications, IoT/cyber-physical systems, and embedded systems. He has successfully graduated 18 Ph.D. students as their principal supervisor, and has co-authored more than 240 journal and conference papers in the aforementioned specialized areas. Dr Loo has been an Associate Editor for *Wiley International Journal of Communication Systems* since 2011. He was the Lead Editor of the book entitled *Mobile Ad Hoc Networks: Current Status and Future Trends* (CRC Press, November 2011).