


RESEARCH

Open Access

# A hotspot-based probabilistic cache placement policy for ICN in MANETs



Cheng Zhang<sup>1,2</sup> , Chunhe Xia<sup>1,2</sup>, Yu Li<sup>1,2</sup>, Haiquan Wang<sup>1,3</sup> and Xiaojian Li<sup>1,4\*</sup>

## Abstract

Cache placement is an integral part of information centric networking (ICN) for optimizing the performance of network. The majority of the existing research on cache placement explores fixed or semi fixed network. The present study proposed a cache placement/update policy in a mobile ad-hoc network (MANET) environment, where the consumers and providers of information are constantly moving. On the basis of request probability and transition probability analysis, a novel approach called cache rebalancing is projected to deal with the cache placement/update problem between different regions. Besides, this study also assesses the performance of the proposed approach in comparison with current caching scheme in large-scale real urban mobility traces, demonstrating that our approach are advantageous in hit ratio, hop count, and other performance indexes over previous strategy. Moreover, the cache steady state of our approach is also obtained to demonstrate the request probability in any region.

**Keywords:** Cache placement policy, ICN, MANET

## 1 Introduction

Due to the lack of effective data access and content distribution measures in the traditional host-centric Internet communication mode, the information-centric network (ICN) has become a new way of thinking. ICN realizes location independence through name-based routing, allows a variety of valuable features, such as mobility support, privacy preserving, multicast, and in-network caching [1–4]. As a result, a more efficient means of communication can be provided, which allows users to re-obtain content more easily. When content is requested by the user, any node that receives the request can respond to the request through a copy. As a result, the bandwidth consumption, access latency, and network congestion can be decreased.

By using name-based routing, ICN naturally equips the application of mobile peer-to-peer communications with highly mobile nodes, lacking steady network topology [5]. In-network caching mechanism confronts the difficulty of placing or distributing contents in a mobile

network, so that the efficiency of the cache network can be enhanced.

Considering the problem of content replication and placement in a mobile ad-hoc network, the method for this article aims to make the optimization of network performance because network nodes are moving continually. Besides, this study also puts forward a placement/update strategy in a MANET environment where both provider and consumer of content are mobile. The model determines whether the efficiency and stability of ICN can be kept by caching or updating popular content data in a MANET environment. To solve this issue, hot city geographic locations and hotspots are considered. The popular data contents can be placed by the proposed model according to user requirements and hotspots. In addition, since taxi moving behaviors show strong association with their geographic features, they are explored for the recognition and division of city hotspot regions [6].

This paper has the contributions as follows:

1. We put forward a novel cache placement and update strategy in a MANET environment where nodes are both data consumers and providers. We argue that geographic information and user

\* Correspondence: [xiaojian@mailbox.gxnu.edu.cn](mailto:xiaojian@mailbox.gxnu.edu.cn)

<sup>1</sup>Beijing Key Laboratory of Network Technology, Beihang University, Beijing, China

<sup>4</sup>College of Computer Science & Information Engineering & College of Software, Guangxi Normal University, Guilin, China

Full list of author information is available at the end of the article

requirements are related, so it is fit to build a hotspot-based caching scheme which provides location-sensitive data contents for users.

2. The proposed cache placement/update policy considers the problem of node mobility within one region and between regions. When node moves within a hotspot, the data request pattern is modeled by introducing a history interest table (HIT). When node travels from one region to another, a cache rebalancing (CR) process is presented to let the node acquire new caches that fit the data requirement of the new location. The approach asks any node about the HIT of the new area it enters, indicating the node can actively acquire the requirement patterns of users. Meanwhile, the request info of the new region is added into the node's own cache in order to rebalance node's caching table.
3. Through inter contact time analysis, we also obtain the cache steady state for our approach, which is the steady data request probability for any region. To further study the steady state, transition probability for taxis between regions is presented to help calculate the final request probability.

## 2 Related works

The existing studies of content replication and placement strategy in ICN mainly focus on fixed networks [7–12] or partially mobility scenarios [13]. The mobility problem for ICN can be separated into two categories: content provider mobility and consumer mobility. In a MANET environment, due to different throughput rates, high mobility of consumer or provider, and dynamic network topology, researchers of ICN should decrease protocol overhead by adjusting light-weighted caching placement strategies to meet the producer or consumer mobility scenario [14]. Paper [15] put forward a content provider mobility scheme in named data networking (NDN), which added a locator to NDN interest package and a mapping system identifier to locator. As a DNS-like service, the mapping system allows users to search for the latest location of the moving content provider. Wei et al. [16] introduced a content consumer mobility ICN scenario, where mobile users moved from site to site and require content from caching servers in every site through a WIFI connection. Lee et al. [5] put forward a location-aided content management architecture for a content-centric MANET, where both provider and consumer of content are moving. The approach binds data to a geographic location to maintain a copy of content in pre-setup geographic boundaries based on GPS location information. Through the active replication of the required content, the availability of data within the boundaries can be maintained effectively. A combination approach of fixed ICN and mobile

DTN is introduced [17] in order to improve the performance of cached contents using a reputation mechanism. The approach ensures an efficient tradeoff between the overall delivery probability and the energy consumption.

In the field of vehicular networks or known as VANETs, current researches mainly focus on the quality of service (QoS) of streaming communications in highway or other constrained mobility scenarios [18]. The work [19] put forward a source mobility solution for vehicular named-data networking (VANDN) by adopting Floating Content and Home Repository schemes together. The approach can maintain distributed content sharing within a certain geographic region. The work [20] studied the source mobility of NDN with a proxy-based mobility support approach named PMNDN. The approach brought a proxy role into NDN to track the mobility of data source and to maintain data reachability. Paper [4] proposed a cache policy based on community similarity and population in an ICN vehicle-to-vehicle scenario. The communities are identified based on their privacy level: public or private vehicles. Paper [21] projected an ICN-based cooperative caching scheme (ICoC) for VANETs. Two social cooperation roles named partner-assisted and courier-assisted are proposed in order to better the quality of experience (QoE) of multimedia streaming performance in an information-centric caching scenario. A series of analyses and simulations are made for ICN-VANETS in two constrained road maps using existing caching solutions. The results show the mobility patterns and geo-constraints in VANETs play a key role in the performance evaluation of caching schemes [6].

Unlike the current work on only one-sided mobility or evenly distributed geolocation, we try to consider the mobility of both consumer and provider. With respect to the influence of different request patterns in urban geo-locations, an ICN caching update and placement strategy is designed for a highly dynamic MANET environment.

## 3 Model overview and specification

The proposed model is designed for the management of content placement/update strategy in a MANET environment, where both provider and consumer of content are mobile. To maintain the efficiency and stability of ICN, the model determines whether or where popular content data should be updated or cached. We make great attempts to deal with this issue with the considerations of hot city geographic locations, or hotspots. Therefore, the popular data contents can be placed in accordance with user requirements and hotspots [22]. Besides, due to the correlation between taxi behaviors and geographic features, they are also taken into account for the recognition and division of city hotspots.

However, the issue for caching placement/update strategy still needs further discussion since people tend to acquire location-sensitive data contents when moving from one region to another. Based on our previous research on data placement/update strategy in a region where users prefer to adopt a similar data requirement model, an inter-region data update strategy is proposed to adapt to the locations of moving user. Moreover, the node is made to update its caching.

Figure 1 shows the general process of our approach we name as cache rebalancing (CR). The two regions possess location-related data contents people are interested in, and nodes can move between them. When node x move from region A to region B, it needs to “re-balance” its cached contents in order to fit in with the requirements of users in region B. To manage this task, node x asks for the history interest table (HIT) of its neighbors so as to calculate the interest distributions for any content data in region B. When node x gets full knowledge of user requirement patterns in region B, it yields interest package for any content in a probabilistic manner with respect to CRE, and caches all the responded content. Node x calculates the caching probability of a data that passes by through  $P_{caching}$  process, which is the update process of CR. The detail of CR process will be further discussed in Section 4.

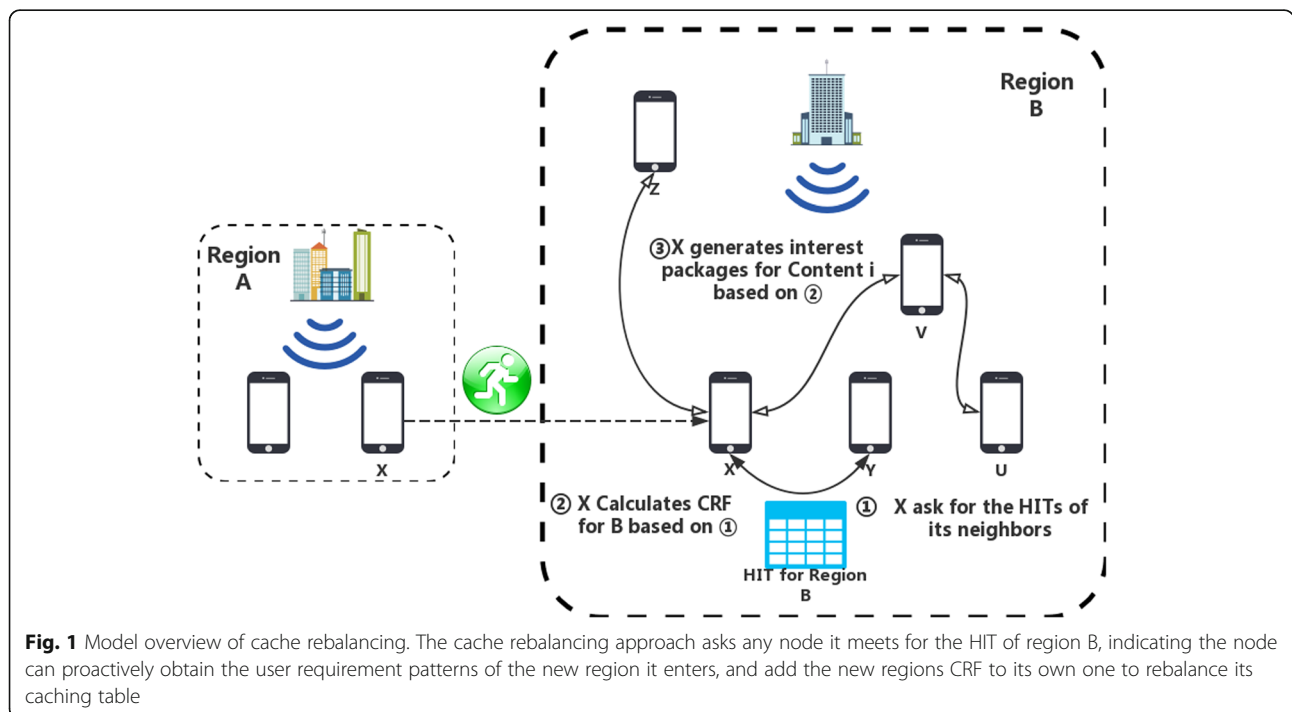
Before we get to the CR analysis, we should take a deeper look at the mobility behaviors of real cities. By analyzing a real-world GPS data generated by over 3000 taxis in Beijing [23], we noticed that taxis’ occupation

status present strong geographic feature. The moving patterns can be very different when a taxi is occupied or carrying idler. Taxi’s status and event can be modeled by Fig. 2.

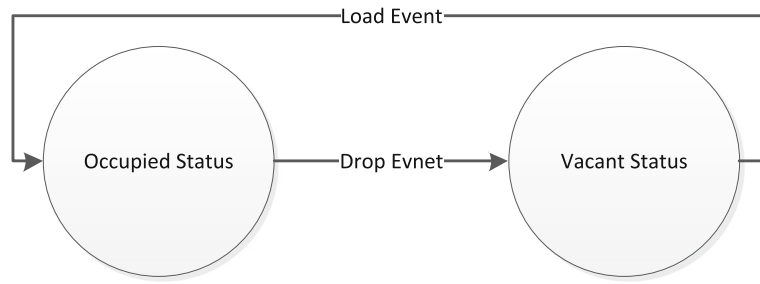
From daily life experience, a taxi tends to move randomly and slowly in order to pick up passengers, while focuses on a destination when it is occupied. Thus we argue that taxi’s moving behavior is more related to people’s geo-interest when it takes a passenger than not. Therefore, in order to better investigate our CR approach, we focus on taxis’ behavior when they are in occupied status. The load and drop events for taxis will be further studied in Section 4 in region transition probability calculations for steady state analysis of CR.

### 4 Problem analysis

In this section, the inter contact time (ICT) between node pairs is discussed in order to get the data update time interval for CR. Then we obtain the steady state formation of the cache network. To calculate  $P_{i,j}$  and  $T_{i,j}$  in the formation, we bring up two schemes: the request probability within region and transition probability between regions. HIT, which is the most suitable representative to denote user query pattern within region, plays an important role in the request probability within region scheme. In transition probability between regions scheme, taxi drop and load events are studied comprehensively in order to calculate the region transition probability for a taxi (Fig. 3).



**Fig. 1** Model overview of cache rebalancing. The cache rebalancing approach asks any node it meets for the HIT of region B, indicating the node can proactively obtain the user requirement patterns of the new region it enters, and add the new regions CRE to its own one to rebalance its caching table



**Fig. 2** Taxi status and event. The moving patterns can be very different when a taxi is occupied or carrying idler. By analyzing the taxi trace dataset, a taxi’s status and event can be modeled by this figure

Through periodically *Cache Rebalancing*, the caching network tends to reach a steady state. Our goal to the CR model is to achieve a cache dynamic stable state [24] for ICN. To achieve that goal, we adopt our previous work showing that the arrival time interval of the request conforms to Poisson distribution [25]:

$$P_n(t) = \frac{1}{n!} (\lambda t)^n e^{-\lambda t} \tag{1}$$

The above distribution function shows the probability for node pair  $x$  encounters  $n$  messages in the time interval  $(0, T]$ , or the CDF of ICT between nodes, where  $\lambda$  is the exponential parameter. We learn that ICTs are non-negative random variables. The event  $\{x > T\}$  means no contact in  $(0, T]$ , and we get  $\{x > T\} = \{N(T) = 0\}$ . As we

already know that  $N(t)$  Possion( $\lambda t$ ) from (4), we can then obtain

$$P(N(T) = 0) = \frac{1}{0!} (\lambda T)^0 e^{-\lambda T} = e^{-\lambda T} \tag{2}$$

Therefore, we can further get the probability of node  $x$  getting at least 1 contact:

$$\begin{aligned} F_{T_i}(T) &= P(T_i \leq T) = 1 - P(T_i > T) \\ &= 1 - P\{N(T) = 0\} = 1 - e^{-\lambda T} \end{aligned} \tag{3}$$

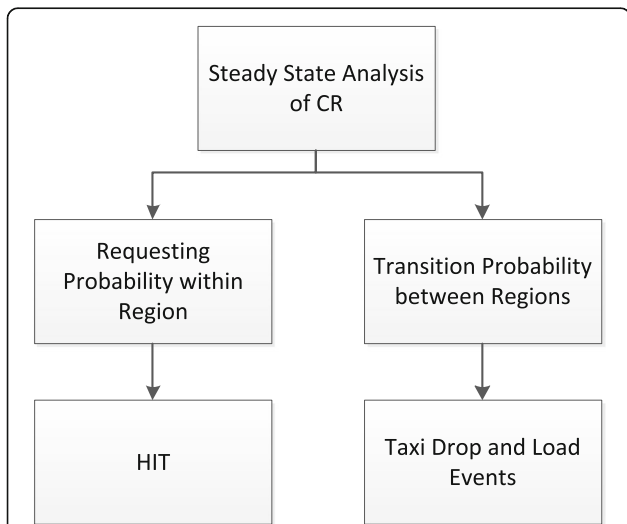
From above, (6) represents the PDF of ICT between nodes A and B. The update interval of data should also adapt to upper distribution. By updating cache periodically, the *Caching Rebalancing* mechanism makes the caching exchanges between various regions cost-free and smooth. Furthermore, the steady state can be obtained for a cache network between regions as follows:

$$R_{r,j} = P_{r,j} + \sum_{i \neq j} R_{r,i} T_{i,j} \tag{4}$$

$P_{r,j}$  denotes the local requesting probability of the  $r$ th ranking content data requested in region  $j$ .  $R_{r,j}$  represents the probability of the  $r$ th ranking data content requested in region  $j$ .  $T_{i,j}$  is the transition probability of node  $x$  moving from  $i$  to  $j$ . Below, we aim to get  $P_{r,j}$  and  $T_{i,j}$  respectfully by request probability within region and transition probability between regions. In order to complete the stable state function, we take HIT into account to calculate  $P_{r,j}$ . On the other hand, taxi drop and load events are brought to calculate the region transition probability  $T_{i,j}$  for a taxi.

**4.1 Request probability within region**

Due to different query patterns in various regions, this part uses HIT to diversity the requirements for content in different regions. Consequently, the strategy should be adapted to different locations for the optimization of data placement/update. In traditional content-centric networking (CCN) solution, which is a realization of ICN, forwarding information base (FIB), pending interest



**Fig. 3** Schemes for steady state analysis. To calculate  $P_{r,j}$  and  $T_{i,j}$  in the formation, we bring up two schemes: the request probability within region and transition probability between regions. In the request probability within region scheme, HIT plays a key role as it is the most suitable representative to denote user query pattern within region. In transition probability between regions scheme, Taxi drop and load events are studied comprehensively in order to calculate the region transition probability for a taxi

table (PIT), and content store (CS) are three critical factors to a node to participate in the network [26]. FIB includes the advertisements and routing information of content, so that interest packages can be forwarded to content providers who may hold the matching data. As PIT tracks interest packages, the matching data from content providers can maintain the reverse path to the data consumers. CS is a storage cache, which allows the temporary caching of the content data by the nodes from different caching schemes. If the data requirement is satisfied in a node, the corresponding entry in PIT will be deleted, which can be found in Fig. 4. Nevertheless, it can be argued that PIT is actually a very important foundation to obtain requirement of users and interest patterns. Therefore, the past PIT entries should not be deleted, but being saved.

From above analysis, HIT is presented to record historical content requirements in a period of time:

$$\text{HIT} \langle \text{Name, Time, Location} \rangle$$

Therefore, HIT records the content name requested by the user, the requesting location, and time of the request. By adding HIT into all ICN nodes, the cache network can now provide the ability to store history information regarding the content data that being requested, with time and location stamp on it. Based on the inspiration of researches from fixed CDNs, we notice that the data access rate follows Zipf-like distribution, which can be used to model content popularity or requesting probability. The requesting probability of the  $r$ th ( $1 \leq r \leq N_c$ ) ranking data from the dataset can be represented as follows:

$$P_r = \frac{1/i^\alpha}{\sum_{n=1}^{N_c} 1/n^\alpha}, r = 1, 2, \dots, N_c \tag{5}$$

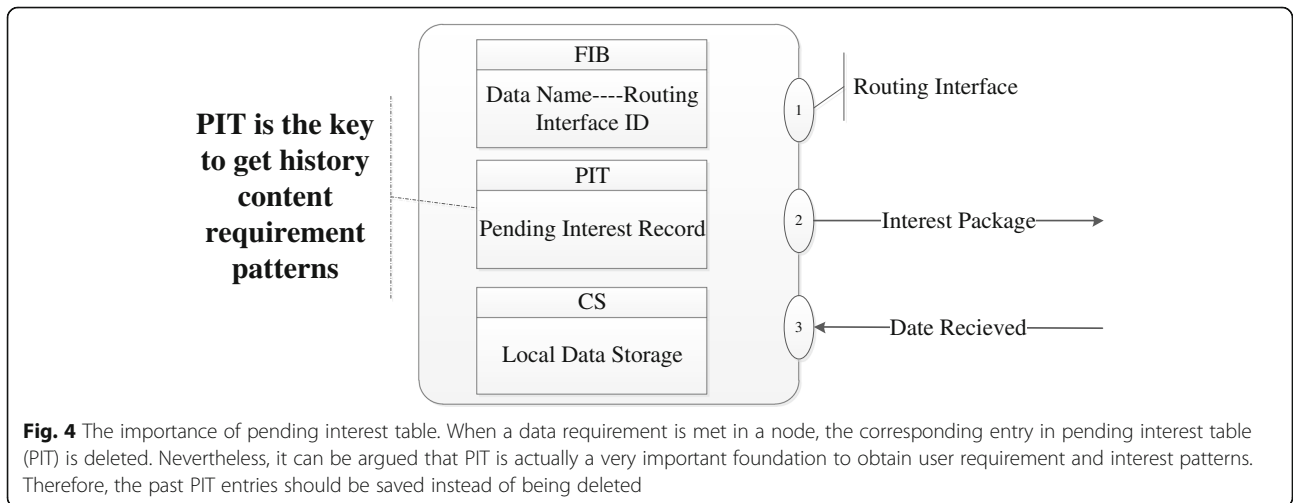
where  $\alpha$  ( $0 \leq \alpha \leq 1$ ) indicates the exponential parameter.

The distribution follows Zipf's law when  $\alpha = 1$ . When  $\alpha = 0$ , uniform distribution is followed.  $N_c$  refers to the number of contents. Therefore, by bringing HIT, a content requirement distribution function CRF can be fitted for a node in a region in terms of Zipf's law:  $P_{r,j}$ , the probability that the  $r$ th ranking content is requested in  $j$ .

In addition to the optimization of data placement, the placement/update strategy also determines whether it is necessary to replace content data, so that the access overhead can be minimized in a dynamic MANET environment. According to Fig. 1, there are mainly two problems: placement/update strategy in a single region and between different regions. Based on the modeling of user requirement with HIT, it can be understood that the requirement pattern is similar in a single region, which is different from different regions. Therefore, we consider dealing with the problem with a *Pcaching* scheme, where HIT, as the most suitable representative for user query pattern in a hotspot, plays the critical role. This scheme considers the data to be cached and the location for caching in a region. Due to the similar query pattern in a single region, a multi-factor scalable probabilistic cache placement strategy considering content popularity, content time-effectiveness, cache occupation, battery status, and HIT is proposed to form a Utility Function  $U$  which takes the above normalized parameters into consideration:

$$U = \sum_{i=1}^{N_p} w_i g(x_i) \tag{6}$$

In the above function, weights  $W_i$  meets  $0 \leq W_i \leq 1$  and  $\sum_{i=1}^N w_i = 1$ , and  $g(x_i)$  is the normalized parameter from cache occupation, battery status, HIT, etc. Since the value of  $U$  is in the interval  $[0:1]$ , the caching probability when a new data package is encountered by a node is obtained: if  $U \rightarrow 1$ , the packet is cached with a



**Fig. 4** The importance of pending interest table. When a data requirement is met in a node, the corresponding entry in pending interest table (PIT) is deleted. Nevertheless, it can be argued that PIT is actually a very important foundation to obtain user requirement and interest patterns. Therefore, the past PIT entries should be saved instead of being deleted

high probability. In comparison, if  $U \rightarrow 0$ , the packet is cached with a low probability. Therefore, when a data packet is received by a node, the current caching probability can be calculated for the packet, which is the *Pcaching* process.

#### 4.2 Transition probability between regions

In this section, we discuss the issue of hotspot recognition transition probability between regions. Taxi drop and load events are brought to calculate the region transition probability  $T_{i,j}$  for a taxi. How to divide and recognize areas are difficulties in current research, and also the basis for calculating area transition matrix. In existing researches, region recognition often simply divide an area into  $m \times n$  mesh; in our case, as the granularity of the mesh grows up, the computation complexity also rises sharply. As for  $m \times n$  mesh, the complexity of calculating area transition matrix is  $O(n^2 \times m^2)$ . Another common way is to divide the area according to a certain attribute (such as cell) manually. This kind of division fit for specific situation, but with low scalability and high subjectivity. Therefore, in order to ensure the precision, computation efficiency, and scalability of the area transition probability, we divide the area into fine-grained meshes and then clustering.

In addition, our area is dynamically divided according to different period of time and different type of events. First of all, we define the following concepts:

**Definition 4.1:** Cell is the smallest unit in area recognition, which specifies the range of a certain area and also represents a set of continuous longitude and latitude. This article defines the cell as a rectangular one, denoted as follows:

$$C_{x,y} := \left\{ \langle lon, lat \rangle \mid x \leq \frac{lon}{len_h} \leq x + 1 \cap y \leq \frac{lat}{len_v} \leq y + 1 \right\} \quad (7)$$

**Definition 4.2:** Region is a set of continuous cells; in this paper, it is also the minimum unit of calculating area transition probability matrix. Denoted as follows:

$$R_m := \{ C_{x,y} \mid \exists C_{i,j} \in R_m \rightarrow \|x-i\| \leq 1 \cap \|y-j\| \leq 1 \} \quad (8)$$

We divide them by the number of events that occurred in the corresponding period. Due to the uneven distribution of events, we also specify two types of areas, one is the event dense area, and the other is the event sparse area.

When the number of events in each cell of area all above the threshold  $\phi_{event}$ , the area is defined as event dense area; otherwise, it is event sparse area. And define the number of event dense area  $\phi_{top}$  for the sake of reducing the number of event dense area that only contains one or a few cells. Meanwhile, we define that an

area contains  $\phi_{size}$  rectangular cells maximally for ensuring the area size not too large, namely

$$\|R_i\| \leq \phi_{size}.$$

As for a specified type of events in a certain period, we calculate the number of events in each cell. First, we need to find out  $\phi_{top}$  event dense area, we start with the densest cell of events in descending order, and conduct breadth traversal. If the number of events is larger than  $\phi_{event}$ , we add it to the area, and each cell only belongs to one area. Once we find the  $\phi_{top}$  area, the number of events in the rest of the cell need not larger than  $\phi_{event}$ ; it just has to be less than or equal to  $\phi_{size}$ .

$\phi_{event}$  is set to be twice of the mean value of the number of events in the corresponding period and event type in this whole scene, and relevant parameters of area recognition are shown in Table 1.

The result of area recognition is shown in Fig. 5, where each color block represents an area.

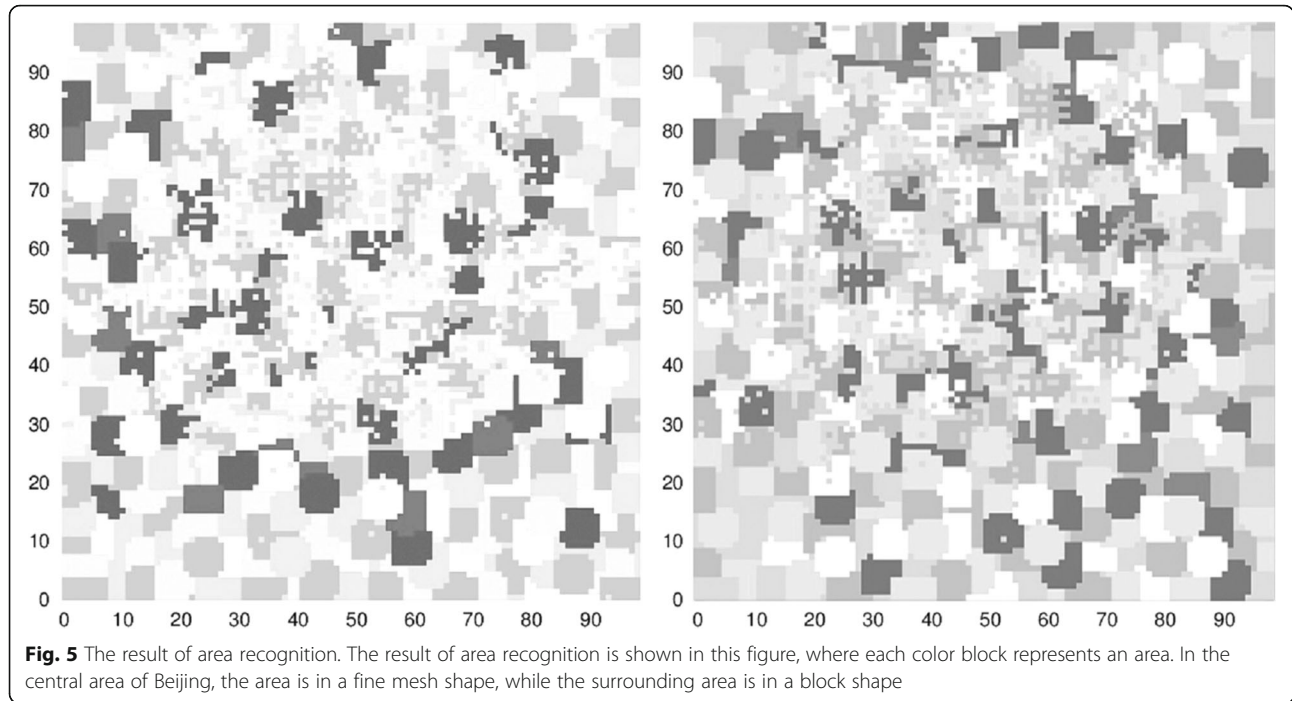
As shown from the above figure, in the central area of Beijing, the area is in a fine mesh shape, while the surrounding area is in a block shape. Such a situation is in line with the expectation, cause the number of events that out of the  $\phi_{top}$  area need not larger than  $\phi_{event}$ . In reality, the distribution of pick-up and drop-off events in Beijing is primarily concentrated on the main road, while other regular areas are sparse and mainly distributed around the city.

The next, based on the divided area, we need to calculate the area transition probability matrix. The area transition probability matrix represents the probability relation of the traffic flow from the starting area to the destination area in a taxi journey. The area transition probability matrix is defined differently according to different time and state transitions, as shown in formulas (9) and (10).

$$P_{load \rightarrow drop}(t) = \begin{pmatrix} P_{load_0 \rightarrow drop_0}^t & P_{load_0 \rightarrow drop_1}^t & \dots & P_{load_0 \rightarrow drop_m}^t \\ P_{load_1 \rightarrow drop_0}^t & P_{load_1 \rightarrow drop_1}^t & \dots & P_{load_1 \rightarrow drop_m}^t \\ \dots & \dots & \dots & \dots \\ P_{load_n \rightarrow drop_0}^t & P_{load_n \rightarrow drop_1}^t & \dots & P_{load_n \rightarrow drop_m}^t \end{pmatrix} \quad (9)$$

**Table 1** Relevant parameters of area recognition

Item/Time	0:00–9:00	9:00–13:00	13:00–21:00	21:00–24:00
$\eta_{load}$	56	84	180	51
$\eta_{load}$	58	84	182	51
$\phi_{size}$	500	500	500	500
$\phi_{top}$	200	200	200	200



$$P_{\text{drop} \rightarrow \text{load}}(t) = \begin{pmatrix} P_{\text{drop}_0 \rightarrow \text{load}_0}^t & P_{\text{drop}_0 \rightarrow \text{load}_1}^t & \dots & P_{\text{drop}_0 \rightarrow \text{load}_n}^t \\ P_{\text{drop}_1 \rightarrow \text{load}_0}^t & P_{\text{drop}_1 \rightarrow \text{load}_1}^t & \dots & P_{\text{drop}_1 \rightarrow \text{load}_n}^t \\ \dots & \dots & \dots & \dots \\ P_{\text{drop}_n \rightarrow \text{load}_0}^t & P_{\text{drop}_n \rightarrow \text{load}_1}^t & \dots & P_{\text{drop}_n \rightarrow \text{load}_n}^t \end{pmatrix} \quad (10)$$

For time  $t$ , the area is divided into the passenger carrying area and the non-load area according to the passenger carrying and disembarkation events. Then the calculation rules from the passenger carrying area to the disembarkation area are shown in formula (4.6):

$$P_{\text{load}_i \rightarrow \text{drop}_j}^t = \frac{\left| \left\{ \text{taxi}_{\text{load}_i}^{t'} \mid t'' > t' \right\} \cap \left\{ \text{taxi}_{\text{load}_j}^{t'} \mid t \leq t' < t + \Delta \right\} \right|}{\left| \left\{ \text{taxi}_{\text{load}_i}^{t'} \mid t \leq t' < t + \Delta \right\} \right|} \quad (11)$$

In which, the denominator is the absolute value of the set of vehicles that have a passenger event in the passenger carrying area  $i$  in time period  $t$ , and the numerator is the absolute value of the set of vehicles that have a passenger event in the next passenger area  $j$  after the passenger carrying events. Therefore, through load and drop events analysis of taxis, we get the transition probability between regions. Furthermore, combined with request probability discussed in 4.1, the steady state function (4) is finally solved.

### 5 Performance evaluation and results discussion

This section firstly gives region recognition results with real city taxi traces of Beijing. We argue that region recognition is the key for us to calculate the transition

probability between regions, and drop event hotspots result is better suited to model people’s moving features. Secondly, based on ns-3, the performance of our CR scheme using ndnSIM [27] simulator is evaluated, within which the taxi traces are used for the optimal simulation of node mobility. We attempt to show the advantages of the proposed caching placement strategy in comparison with the current ICN cache placement approach leave copy everywhere (LCE), which is the default cache placement policy of CCN (Table 2).

#### 5.1 Hotspot recognition and division

The goal of this section is to evaluate the strong connection between taxi moving behavior and geographic feature. We argue that the distributions of taxi load and drop actions both have significant geographic preferences. In the simulation, we analyze taxis’ status and event changes, along with time, latitude and

**Table 2** Simulation parameters

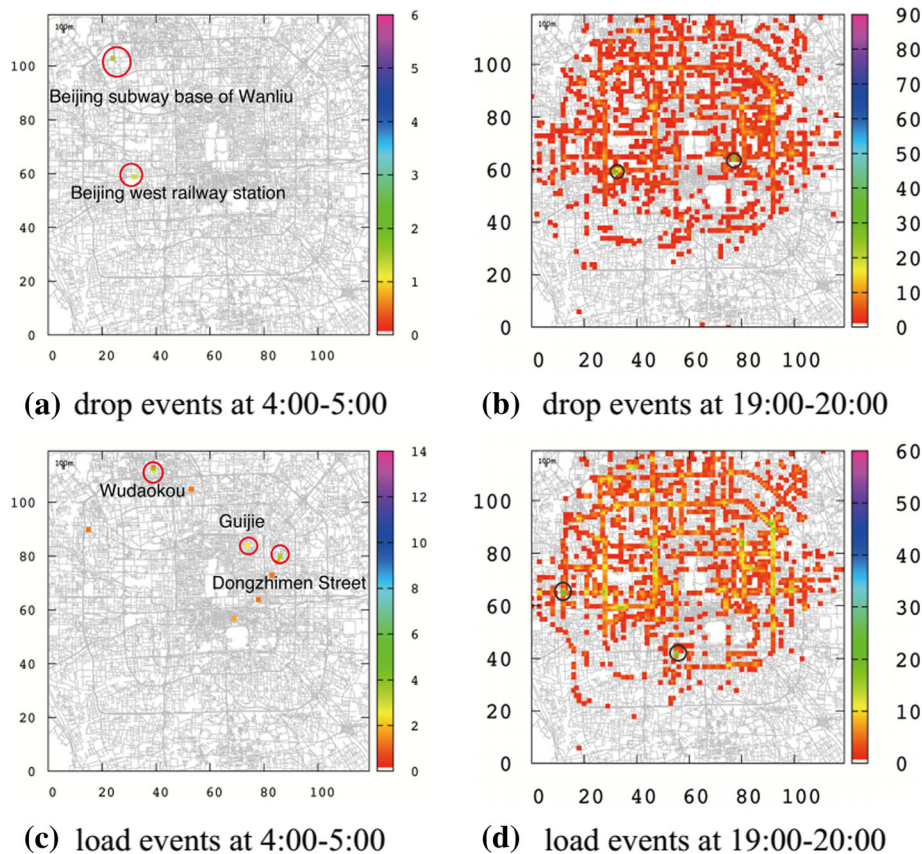
Parameter	Value
Area size	24 km × 24 km
Time of simulation	7200 s
Number of nodes	3000
Node number range	(500, 3000)
Radio range	250 m
Zipf (q, a)	(0, 1)
Node speed	10 m/s
Speed range	(2 m/s, 20 m/s)

longitude, and speed information. The data was analyzed for a week of November 2011, by evaluating the data quality and eliminating the special features of the holiday.

Figure 6 denotes the hotspot recognition results in 4:00–5:00 and 19:00–20:00 for drop events and load ones. From 4 o'clock to 5 o'clock in Fig. 6c, the passenger load incidents mainly occurred in a few locations such as Wudaokou, Guijie Street and Dongzhimenwai Street, and the drop-off incident in Fig. 6a occurred at Beijing Wanliu Operation Station and Beijing West Railway Station. Among them, the Wudaokou area has transportation hubs such as busses and subway stations, and there are clear geographical landmarks like commercial areas, office buildings, and university around it. The Guijie Street is a famous leisure and entertainment area in Beijing. These above areas have high attractions to people and Wanliu's Operation Center turns out to be a taxi operation center where taxis report to work every morning. Beijing West Railway Station fits the scenario of people "catch the train" in the early hours of the morning. Therefore, it is reasonable these areas are already hotspots

early at 4–5 o'clock in the morning. The 19:00–20:00 shown in Fig. 6b, d is a period of high incidence that happens for taxi taking and dropping passengers; the results show that the recognized hotspots distribution is very consistent with the actual main roads in Beijing.

Compare the hotspots for drop and load actions in different time period, we find that the passenger-taking event distribution is more average than the no-load event, and hotspots are repeatable in workday and weekend although the number of events occurred varies. This phenomenon may occur because the passenger-taking incidents are mainly distributed in the residents' homes, while the drop-off locations tend to gather at work places, subway stations, or scenic spots. In this case, drop event hotspots result should be considered as a better candidate to model people's moving features. We conclude our finding in hotspot recognition and division, that is, taxi behavior is significantly related to vehicle status, time and geographical features. Drop events hotspots are more suited for transition probability calculations since they are direct reflection of the passenger's moving patterns.



**Fig. 6** Drop and load events in a city. Based on analyzing drop and load incidents, accurate hotspots of drop and load event happens for taxi can be finally obtained as shown in this figure, (a) and (b) show the drop events recognition results at 4:00-5:00 and 19:00-20:00, (c) and (d) show load events results at 4:00-5:00 and 19:00-20:00



## 5.2 Evaluation results and discussion for CR

To prove the benefits of our cache placement approach, we separate our simulation into two parts. The *Pcaching* evaluation part is to show the benefits of our approach, focusing on where the data should be cached within one region; and the CR evaluation demonstrates our advantage on inter-region scenario than existing cache placement policy, mainly deals with the problem of how to maintain a balanced caching state when nodes move from one region to another where people in it have different data interest patterns.

### 5.2.1 *Pcaching* evaluation

*Pcaching* is compared to the default CCN caching strategy leave copy everywhere (LCE) in both cache-based metrics and network-based metrics [2]. The cache-based metrics measure the effectiveness of caching policies through calculating whether a policy is capable of caching and maintaining the desired content. In most cases, metrics based on cache are calculated per node.

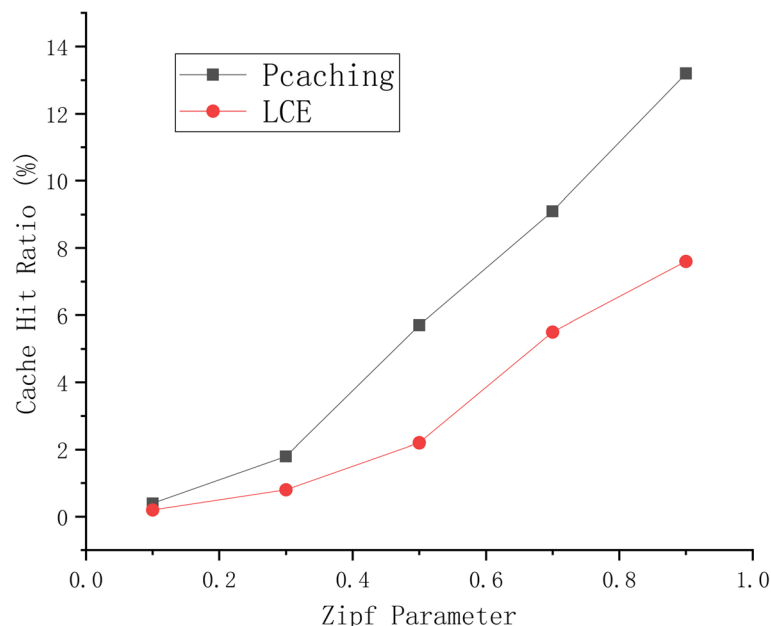
**5.2.1.1 Cache hits** According to Fig. 7, which shows the cache hit ratio if Zipf parameter  $\alpha$  differs from cache placement strategies. According to the large value of Zipf parameter  $\alpha$ , the popularity of a given content is higher than others in the region. Obviously, *Pcaching* presents better cache hit ratio which is about three times of LCE. Therefore, the distribution of workload of content servers in *Pcaching* is better than LCE. The results are not surprising because the LCE policy actively caches

the content block without any choice, which may result in cache redundancy across the network, thereby reducing the cache hit ratio of the network. The cache hit ratio in *Pcaching* is in direct proportion to Zipf parameter, suggesting that the competitive advantage of *Pcaching* comes from the areas where users socialize and share popular content through caching networks.

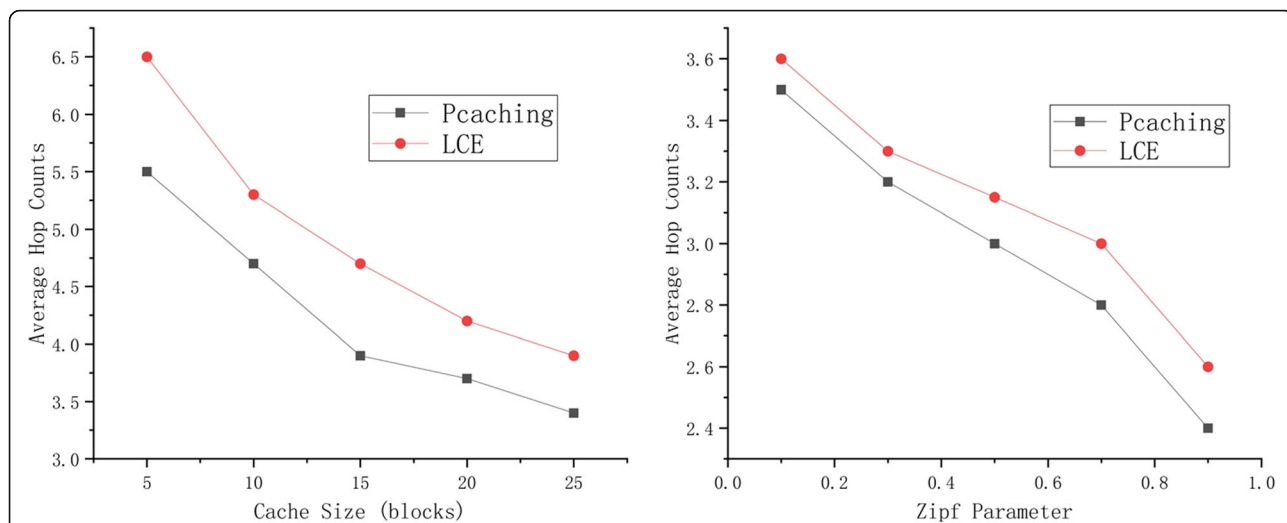
**5.2.1.2 Hop counts** In Fig. 8, the cache sizes of the node ranges from 5 to 25 blocks, accounting for 5% to 25% of total content number. Compared with other scheme LCE, the average hop counts offered by *Pcaching* for different cache sizes is lower. Therefore, the *Pcaching* enhance the efficiency of the whole network by using the cache memory. In addition, it can also be seen that average hop counts decreases with the increase of Zipf parameter, while *Pcaching* is still advantageous to LCE. The results demonstrate *Pcaching* can decrease the hop counts, and *Pcaching* has better performance in regions where social communication patterns are frequent.

### 5.2.2 CR evaluation

After the evaluation of *Pcaching*, which mainly works well within one region, we take a further step on studying *Cache Rebalancing* between regions. The main issue for caching policy in regions is how to achieve a balanced state after a period of time. We study the download time it takes for *Cache Rebalancing* to obtain a cached content under different circumstances by calculating *Hop Counts*. Meanwhile, *Caching efficiency* for



**Fig. 7** Cache hit ratio comparison within one region. This figure shows that *Pcaching* presents better cache hit ratio that is nearly twice higher than LCE, suggesting that workload of content servers in *Pcaching* is better distributed than LCE

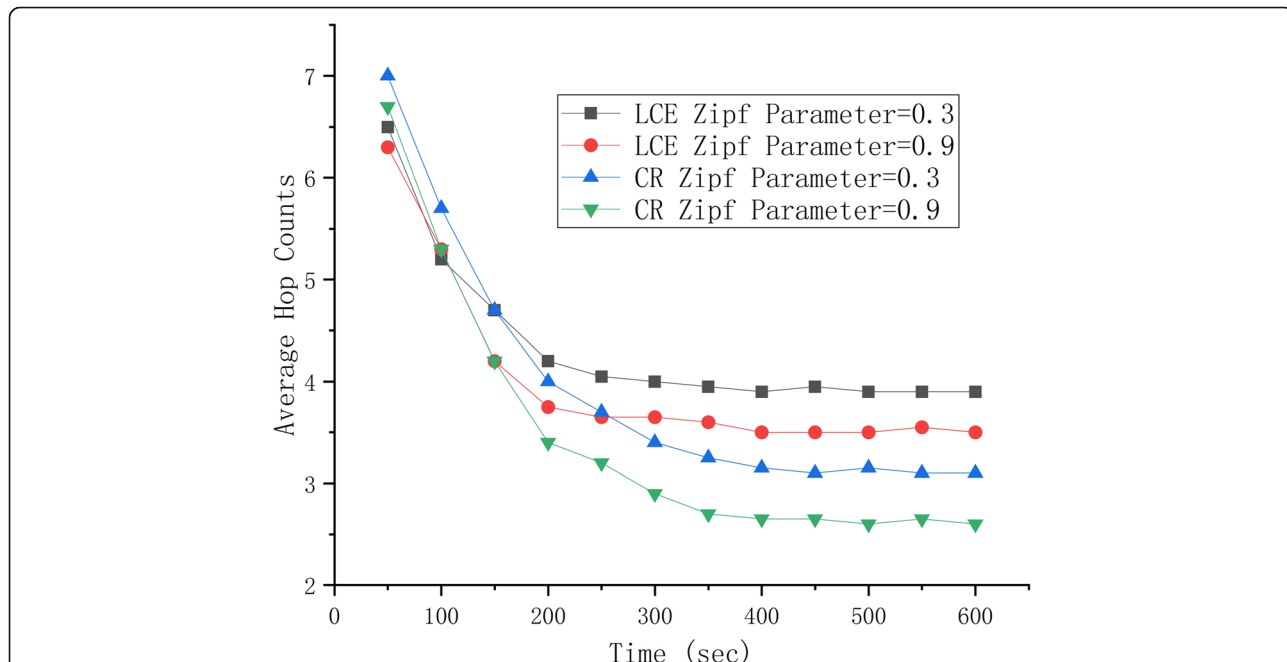


**Fig. 8** Hop counts for comparison within one region. In comparison with the other scheme LCE, we can see that *Pcaching* offers lower average hop counts for different cache sizes, telling that the *Pcaching* efficiently employs the cache memory to enhance the effectiveness of the whole network

different time, node speed, and number of nodes are also evaluated to show the effectiveness of our caching policy.

Figure 9 compares the average hop counts of our approach cache rebalancing (CR) and LCE under different Zipf parameters. The results clearly show that CR outperforms LCE after they reach steady state, which is not surprising as CR takes advantage of hotspot user requirement patterns caching more popular

contents in the nearby storage nodes. Note that CR takes longer time than LCE to reach stable state. We argue that it is mainly because CR is a probabilistic approach, so it needs more time to store enough contents for a better and steady performance. LCE can start operation quickly as it caches every content copy passing by; however, the performance of it is constrained as it lacks of the ability to identify user interests between regions.



**Fig. 9** Hop counts vs. time for cross regions. The results clearly show that CR outperforms LCE after they reach steady state, which is not surprising as CR takes advantage of hotspot user requirement patterns caching more popular contents in the nearby storage nodes. Note that CR takes longer time than LCE to reach stable state

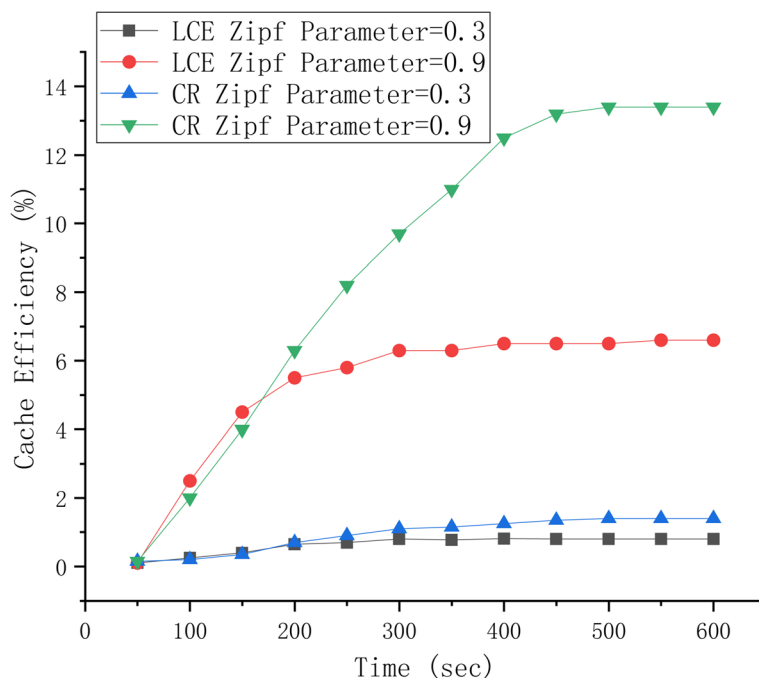
**5.2.2.1 Caching efficiency** Caching efficiency metric is the number of content requests met by a cache and measured as the fraction of the cache hits on a node to the number of contents stored in the cache of the same node. As shown in Fig. 10, CR obviously leads LCE in cache hit ratio for both Zipf Parameters. The results also show that CR has more advantage when Zipf parameter is set to 0.9 rather than 0.3. In the 0.9 case, CR more than doubled the cache hit ratio of LCE, indicating that CR provides a better solution to cross-region scenario where user interests differ from one region to another. Meanwhile, LCE takes no consideration for content requirement, so it lacks the ability to adjust when someone needs to retrieve a data copy from another region. The same thing we find here is CR takes more time to “heat up” than LCE. It is understandable because data are cached by LCE aggressively while CR takes a probabilistic approach to selectively cache popular data.

Figure 11 demonstrates the cache efficiency with different node speed settings when the network reaches stable state. We can see that these two caching schemes perform very differently when facing mobility. For LCE, node moving speed apparently affected the caching performance significantly; the reason for that is LCE needs more “connecting opportunities” to other nodes so that it can exchange latest content copies from other nodes, the increasing node speed gives it just that. On the other hand, CR is designed to address the issue of cross-region

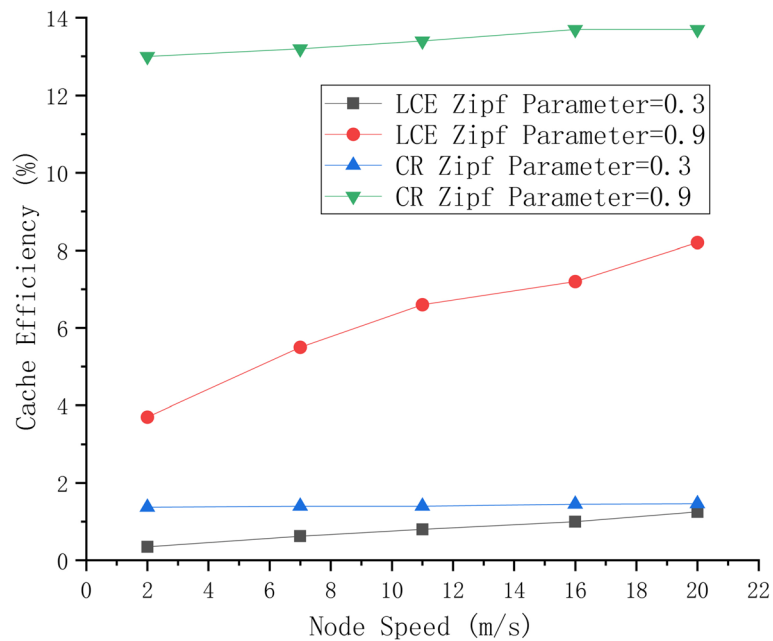
caching, and most of the popular contents are stored in local region nodes according to user requirement model. In that case, node mobility would not affect too much on the performance of CR, which provides a steady caching solution when facing cross-region and mobility problems.

Figure 12 plots the cache efficiency with respect to number of nodes in the whole network. Behind the obvious result that both caching policies benefit from the increasing of number nodes, there are some notable remarks that CR provides a more stable performance in lower number nodes scenario compared to higher ones than that of LCE. When Zipf parameter is set to 0.9, the cache efficiencies for CR in 500 and 3000 nodes are 10.3% and 13.4%, while for LCE they are 3% and 6.6%. The reason is mainly that CR caches popular content copies in a probabilistic pattern for every node, so even in a sparse environment where number of nodes is limited, CR can still function well to provide content to the interests of users. On the contrary, LCE relies on more nodes to cache and deliver content in order to provide a better service.

To summarize this section, we take two steps to design and perform our evaluation on our proposed cache policy. First we setup our simulation environment by hotspot recognition and division using real city taxi traces. Top ranking geo-clusters of taxi drop events happen are picked to be hotspots in a city. With the hotspot



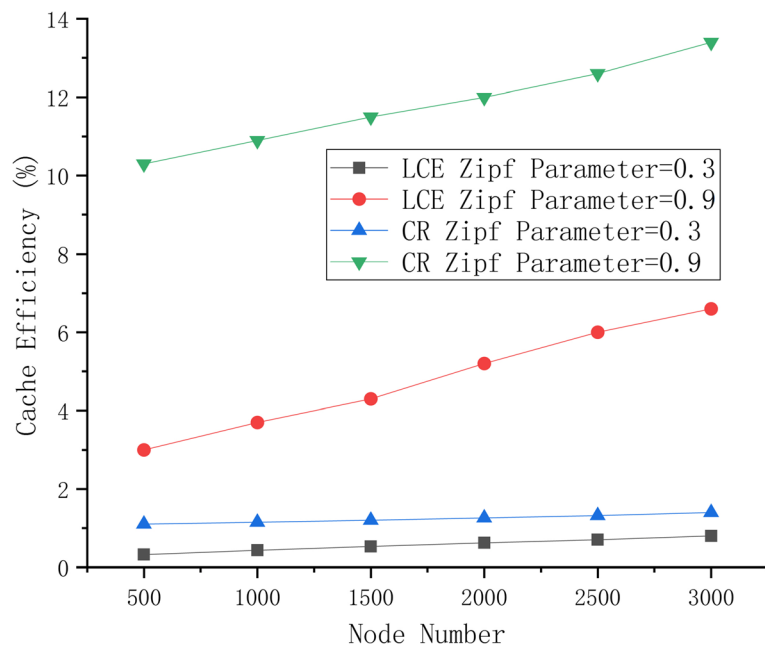
**Fig. 10** Cache efficiency vs. time for cross regions. This figure indicates that CR leads LCE in cache hit ratio for both Zipf parameters. The results also show that CR has more advantage when Zipf Parameter is set to 0.9 rather than 0.3. In the 0.9 case, CR more than doubled the cache hit ratio of LCE, indicating that CR provides a better solution to cross-region scenario where user interests differ from one region to another



**Fig. 11** Cache efficiency vs. node speed. The figure demonstrates the cache efficiency with different node speed settings when the network reaches stable state. The results show that these two caching schemes perform very differently when facing mobility

division results, we secondly evaluate the performance of *Pcaching* and *Cache Rebalancing* in a separate way. The results show that *Pcaching* outperforms LCE within one region in both hop counts and cache hit ratio because users generally socialize and share popular contents through the caching network in a hot region.

*Cache Rebalancing* also presents well results in cross-region scenario, showing steady performance in cache efficiency and hop counts. The probabilistic and socialized feature makes CR more suitable to hotspot and real social environment where user tends to acquire popular data in a geo-related manner. Note that CR might



**Fig. 12** Cache efficiency vs. number of nodes. The figure plots the cache efficiency with respect to number of nodes in the whole network. Results show that CR provides a more stable performance in lower number nodes scenario compared to higher ones than that of LCE

take a longer time to “heat up” than other approaches, however the better outcomes definitely worth the wait.

## 6 Conclusions

To conclude, this work studies cache placement problem for ICN in a MANET environment. On the basis of user request modeling and hotspot recognition, a hotspot-based caching scheme which provides location-sensitive data contents for users are put forward to satisfy the caching placement/update requirements in a single region and between regions. The cache steady state of a cache network between regions is also proposed by adopting the analysis of ICT between nodes. Real city taxi mobility trace simulations demonstrate that taxi behavior is significantly related to vehicle status, time and geographical features, and drop events hotspots are more suited for transition probability calculations since they are direct reflection of the passenger’s moving patterns. The results also indicate that the proposed approach is superior to the traditional CCN caching scheme in both one-region and cross-region scenarios, while reducing network overhead. The solid performance and the socialized feature make our approach more suitable to real social mobility environment where user tends to acquire popular data in a geo-related way.

### Abbreviations

CR: Cache rebalancing; CS: Content store; FIB: Forwarding information base; HIT: History interest table; LCE: Leave copy everywhere; PIT: Pending interest table

### Acknowledgements

Not applicable.

### Funding

This work is supported by these projects: The National Natural Science Foundation Project of China under Grant No. U1636208 and 61862008; the Co-Funding Project of Beijing Municipal Education Commission under Grant No.JD100060630.

### Availability of data and materials

The dataset used in this article is mainly from the traffic regulation authority of Beijing from one of our ongoing project. Due to regulatory reasons, the taxi trace dataset cannot be published online at the current time.

### Authors’ contributions

CZ carried out the modeling of *Pcaching* and *Cache Rebalancing*. CX, HW, and XL participated in the design of the study. CZ and YL performed the evaluation and statistical analysis. All authors read and approved the final manuscript.

### Authors’ information

Not applicable.

### Competing interests

The authors declare that they have no competing interests.

## Publisher’s Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

### Author details

<sup>1</sup>Beijing Key Laboratory of Network Technology, Beihang University, Beijing, China. <sup>2</sup>School of Computer Science and Engineering, Beihang University, Beijing, China. <sup>3</sup>College of Software, Beihang University, Beijing, China.

<sup>4</sup>College of Computer Science & Information Engineering & College of Software, Guangxi Normal University, Guilin, China.

Received: 20 March 2019 Accepted: 3 May 2019

Published online: 27 May 2019

## References

1. Y. Ma, Y. Wu, J. Li, and J. Ge, APCN: A Scalable Architecture for Balancing Accountability and Privacy in Large-scale Content-based Networks, *Information Sciences*, <https://doi.org/10.1016/j.ins.2019.01.054>, In Press
2. A. Ioannou, S. Weber, A survey of caching policies and forwarding mechanisms in information-centric networking. *IEEE Commun. Surv. Tutorials* **18**, 2847 (2016)
3. G. Xylomenos, C.N. Ververidis, V.A. Siris, N. Fotiou, C. Tsilopoulos, X. Vasilakos, K.V. Katsaros, G.C. Polyzos, A survey of information-centric networking research. *IEEE Commun. Surv. Tutorials* **16**, 1024 (2014)
4. W. Zhao, Y. Qin, D. Gao, C.H. Foh, H.C. Chao, An efficient cache strategy in information centric networking vehicle-to-vehicle scenario. *IEEE Access* **5**, 12657 (2017)
5. S. Lee, S. H. Y. Wong, K. Lee and S. Lu, Content management in a mobile ad hoc network: Beyond opportunistic strategy. 2011 Proceedings IEEE INFOCOM, Shanghai, 2011, pp. 266–270
6. F.M. Modesto, A. Boukerche, in *IEEE International Conference on Communications*. An analysis of caching in information-centric vehicular networks (2017)
7. B. Ko, D. Rubenstein, Distributed self-stabilizing placement of replicated resources in emerging networks. *IEEE/ACM Trans. Networking* **13**, 476 (2005)
8. Lili Qiu, V. N. Padmanabhan and G. M. Voelker, On the placement of Web server replicas, Proceedings IEEE INFOCOM 2001. Conference on Computer Communications. Twentieth Annual Joint Conference of the IEEE Computer and Communications Society (Cat. No.01CH37213), Anchorage, AK, USA, 2001, pp. 1587–1596 vol.3
9. E. Cohen, S. Shenker, Replication strategies in unstructured peer-to-peer networks. *Acm Sigcomm Comput. Commun. Rev.* **32**, 177 (2002)
10. Z. Meng, H. Luo, H. Zhang, A survey of caching mechanisms in information-centric networking. *IEEE Commun. Surv. Tutorials* **17**, 1 (2015)
11. Renchao, Hengyang, Zhang, Huang, Yunjie, What to cache: differentiated caching resource allocation and management in information-centric networking. *China Commun.* **13**, 261 (2016)
12. X. Jiang, B.I. Jun, G. Nan, L.I. Zhaogeng, A survey on information-centric networking: Rationales, designs and debates. *China Commun.* **12**, 1 (2015)
13. Z. Yu, A. Afanasyev, J. Burke, L. Zhang, in *Computer Communications Workshops*, vol. A survey of mobility support in named data networking (2016)
14. B. Feng, H. Zhou, X. Qi, Mobility support in named data networking: a survey. *EURASIP J. Wirel. Commun. Netw.* **2016**, 220 (2016)
15. X. Jiang, J. Bi, Y. Wang, P. Lin, Z. Li, in *IEEE International Conference on Network Protocols*. A content provider mobility solution of named data networking, vol 1 (2012)
16. T. Wei, L. Chang, B. Yu, J. Pan, in *Global Communications Conference*. MPCs: a mobility/popularity-based caching strategy for information-centric networks, vol 4629 (2015)
17. S. Arabi, E. Sabir, H. Elbiaze, *Information-centric networking meets delay tolerant networking: beyond edge caching* (2018)
18. Y. Zhou, F. R. Yu, J. Chen and Y. Kuo, Resource Allocation for Information-Centric Virtualized Heterogeneous Networks With In-Network Caching and Mobile Edge Computing. in *IEEE Transactions on Vehicular Technology*, vol. 66, no. 12, pp. 11339–11351, 2017.
19. J.M. Duarte, T. Braun, L.A. Villas, in *Ad Hoc Networks. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, Vol. Source mobility in vehicular named-data networking: An overview (Springer, Cham, 2018), p. 223
20. D. Gao, R. Ying, C. Foh, H. Zhang, A.V. Vasilakos, PMNDN: proxy based mobility support approach in mobile NDN environment. *IEEE Trans. Netw. Serv. Manag.* **14**, 191 (2017)
21. Q. Wei, C. Xu, J. Guan, H. Zhang, L.A. Grieco, in *World of Wireless, Mobile & Multimedia Networks*, vol. Social cooperation for information-centric multimedia streaming in highway VANETs (2014)
22. C. Zhang, C. Xia, H. Wang, in 228, ed. by Y. Sun, T. Lu, X. Xie, L. Gao, H. Fan. A probabilistic and rebalancing cache placement strategy for ICN in MANETs (Springer Singapore, Singapore, 2019)
23. H. Wang, W. Yang, J. Zhang, J. Zhao, Y. Wang, in *Computer Communications Workshops*, vol. START: status and region aware taxi mobility model for urban vehicular networks (2015), p. 594

24. E. J. Rosensweig, D. S. Menasche and J. Kurose, *On the steady-state of cache networks*. 2013 Proceedings IEEE INFOCOM, Turin, 2013, pp. 863–871
25. Y. Hu, H. Wang, C. Xia, W. Li, Y. Ying, in *IEEE Conference on Local Computer Networks*, vol. On the distribution of inter contact time for DTNs (2012)
26. W. You, B. Mathieu, G. Simon, in *Network of the Future*. How to make content-centric networks interwork with CDN networks, vol 1 (2014)
27. A. Afanasyev, I. Moiseenko, L. Zhang, *ndnSIM: ndn simulator for NS-3* (2012)

**Submit your manuscript to a SpringerOpen<sup>®</sup> journal and benefit from:**

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

---

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)

---