文章编号：1007-5321(2020)01-0080-12　　DOI：10. 13190/j. jbupt. 2019-054

基于本地内容流行度预测的主动缓存策略

任佳智，　田　辉，　聂高峰
（北京邮电大学 网络与交换技术国家重点实验室，北京 100876）

摘要：已有边缘缓存技术研究假设已知固定的全局流行度，忽略了反映基站接收到的内容请求历史中的流行度地域差异特性和动态特性，为此，提出了一种基于本地内容流行度预测的内容部署策略. 首先，考虑流行度的地域特性，将内容请求历史记录相似的小基站分簇；然后，使用线性回归方法预测每个小基站簇群的本地内容流行度，基于预测的本地内容流行度，利用随机几何和凸优化理论求得次优内容部署决策；最后，基于真实数据集的实验验证了所提算法性能以及相应的缓存系统性能. 仿真结果表明，所提算法优于对比算法的缓存命中率性能.

关　键　词：内容部署；本地流行度预测；余弦相似性
中图分类号：TN929. 53　　**文献标志码：**A

Proactive Caching Scheme with Local Content Popularity Prediction

REN Jia-zhi，　TIAN Hui，　NIE Gao-feng
(State Key Laboratory of Networking and Switch Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China)

Abstract: Considering the problem that most works on content placement so far consider global popularity, neglecting the demand difference between base stations (BSs), a content placement scheme based on similarity between small base stations (SBSs) and local content popularity prediction considering popularities' geographical diversity are proposed. Firstly, SBSs that possess similar historical content requests is identified by similarity measurements. Then the probabilities of future requests are predicted for each similar SBS group utilizing linear regression method. Based on this local popularity, the sub-optimal content placement decision is made according to stochastic geometry and convex optimization. Thereafter, real data sets to verify our prediction algorithm and investigate system performance are used. It is shown that the proposed scheme outperforms the comparison schemes in terms of hit ratio.

Key words: content placement; local popularity prediction; cosine similarity

More and more data traffic are coming from cellular networks in the last decade. One of the incentives is the proliferation of smart mobile devices, which are equipped with advanced multimedia capabilities. As of today, mobile internet users can enjoy high data rate applications, such as high-definition video service, thanks to the evolution of cellular networks. According to report released by Cisco, video on demand service is anticipated to keep growing and will account for the majority of total mobile data traffic in a couple of

收稿日期：2019-04-10
基金项目：国家技术重大专项项目(2018ZX03001019-003)；国家自然科学基金青年基金项目(61801044)
作者简介：任佳智(1987—)，男，博士生.
通信作者：田　辉(1963—)，女，教授，博士生导师，E-mail：tianhui@ bupt. edu. cn.

years[1]. Nevertheless, the capacity increase in cellular networks thanks to application of new technologies cannot match with the even faster growth of mobile data traffic[2]. Encouraged by the success of content caching in computer networks, edge caching technique is envisioned to play a key role in future wireless content delivery[3]. This is inspired by that fact that in real life only a small portion of most popular contents are requested frequently by the majority of mobile users[4].

Content placement is one of the fundamental problems for the integration of edge caching technique into future cellular networks. The design of efficient caching policies usually aims at the optimization for system metrics of total hit ratio[5], network throughput[6] and energy efficiency[7]. Caching contents as close as possible to the requester is able to reduce service delay and achieve higher energy efficiency. In addition, placing as many contents as possible at the storage of edge nodes can obviously result in higher total hit ratio. However, storage capacity, processing power, battery lifetime and many other kinds of computation or communication resources for edge nodes are limited. This constraint on edge node resources will naturally impact the system performance gain brought by application of caching technique due to the increasing of content variety and quality requirement. Thus, efficient caching schemes in a proactive manner, addressing the problems of where to cache, what to cache and how to cache, must be thoroughly studied.

Wireless connection and content popularity model are two vital elements in the design of efficient caching policies. So far, a lot of studies have been done based on these two factors. Kiskani et al. [8] study decentralized coded content caching for next generation cellular networks. The contents are linearly combined and cached in under-utilized caches of user terminals and its throughput capacity is compared with decentralized un-coded content caching. Liu et al. [9] consider a commercialized small-cell caching system consisting of a network service provider (NSP), several content providers (CPs), and multiple mobile users (MUs). One cooperative network with caching relays to reduce the transmission links overheard by the eavesdropper is proposed[10] in order to enhance the physical layer security. Chen et al. [11] consider two kinds of network architectures, always-on architecture and the dynamic on-off architecture, to study a probabilistic small-cell caching strategy, where each SBS caches a subset of contents with a specific caching probability.

Most of these works perform content placement assuming identical content popularity. In other words, the request probability of each file is assumed to be identical for every user. In fact, identical content popularity may not exist due to diverse personal backgrounds and social relations. Therefore, it is not suitable for probabilistic wireless caching problem. Zeng et al. [12] design an edge caching strategy for app services based on the observed characteristics of BSs according to real data collected by network operator. Some papers leverage machine learning and collaborative filtering to estimate content popularity for the proactive caching scheme[13]. A social-driven propagation dynamics-based prediction model, which requires neither training phases nor prior knowledge is proposed[14]. These studies provide new insights on proactive caching. Nevertheless, there is no widely accepted approach to construct prediction models for the proactive caching problem assuming insufficient knowledge of the content popularity matrix.

In this paper, we propose a content placement scheme based on local popularity prediction. Firstly, small base stations that possess similar historical content requests are identified by similarity measurements. Then the probabilities of future requests are predicted for each similar SBS group utilizing various prediction methods. Based on this local popularity, optimal content placement decision is made according to stochastic geometry and convex optimization. Moreover, we use real dataset to verify our prediction algorithm and generate data to evaluate the proposed algorithm. More specifically, we use arithmetic mean prediction as the baseline algorithm. This arithmetic mean prediction method is linear regression with equal parameters. The fact that our proposed algorithm outperforms the baseline verifies the assumption of unequal linear correlation between past and future content popularity.

1 System model and problem

1.1 System model

A typical scenario of the cellular network is illustrated in Fig. 1. As shown in the figure, the coverage area of the small base stations (SBSs) consists of many types of regions, such as university campus, office building, shopping mall and housing estate. The backhaul link between SBSs and the macro base station (MBS) is wireless and the placing phase of contents is executed in idle hours. The MBS can access all the items of CP via high-speed link such as fiber. We assume that an individual user stays in only one region, such as campus. In other words, each user is able to connect with a fixed subset of all the SBSs. This can be verified by the fact that in real life an individual user usually stays at a fixed site, such as home or workplace. And users' occasional movement to other regions will not affect the study in this paper, since we focus on the design of content placement on a relatively large time-scale. The edge nodes in each area are deployed based on an independent homogeneous Poisson point process with intensity λ_s and λ_u for SBSs and users, respectively.

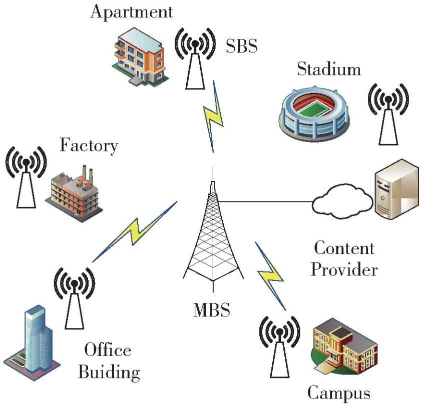

Fig. 1 System model for local popularity prediction

We consider a cellular network, where N_u wireless users intend to retrieve interested files from an MBS. In our model, the library of available files is a finite set, which is denoted by $F = \{f_1, f_2, \cdots, f_N\}$ and stored in the memory of the MBS. Each file has the equal size of B bit. This paper will not consider cases of different file sizes. Since each file can always be divided into pieces of same size, this assumption does not affect the result of this study. It is supposed that each user at a time has a request for an entire file from the library. In order to bring contents closer to users, N_s SBSs are deployed in the coverage area of the MBS. Each SBS has a memory capacity of MB bit that is at most M entire files can be cached in SBS storage and M is assumed to be less than library size N.

Each file is requested by the users with a probability. Specifically, we refer to the probability as the file popularity and denote the popularity of all the files by aset $P = \{p_1, p_2, \cdots, p_N\}$, where file j is requested with probability p_j and $\sum_{j=1}^{N} p_j = 1$. Due to interest diversity of users, users' file popularity under different SBSs' coverage area may be very different. As a result, users' file popularity measured at a single SBS might be different from the global popularity measured at the MBS.

We consider the signal propagation-loss model, which is the combination of deterministic path loss and random log-normal shadowing. Accordingly, the signal-to-interference-plus-noise ratio (SINR) of the received signal is derived as

$$X_i = \frac{P_t S_i / L_i}{P_t \sum_{j \neq i} S_j / L_j + N_0} \tag{1}$$

where P_t is the transmission power, S_i is the shadowing experienced by user and L_i is the path loss. The constant N_0 is the noise power and $P_t \sum_{j \neq i} S_j / L_j$ is the interference from the SBSs other than the connected SBS.

Any SBS, which can provide the user a higher SINR than the default SINR threshold δ, is supposed to be able to deliver the requested file. Given the minimal required transmitting rate for contents and the available bandwidth, we can derive the threshold from the classical Shannon channel capacity formula. When a user requests for a particular file, it will firstly try to retrieve the content from one of the available SBSs. If the desired file is not present in any covering SBSs' cache, the MBS will transmit the content to the user.

1.2 Problem formulation

To achieve maximum quality from edge caching, we choose to maximize the total hit ratio, with constraints on storage capacity. This total hit ratio is defined as the probability that a typical user is able to retrieve the requested content from one of the covering SBSs. This occurs only when the user is covered by at least one SBS and the requested content is stored in the cache memory by at least one of the covering SBSs.

We denote the content placement for all SBSs by an $N_s \times N$ matrix I in which $I_{s,n}$ is the element at sth row and nth column taking value from $\{0,1\}$. $I_{s,n}=1$ if sth SBS caches nth file, and 0 otherwise. The content requests of all users are denoted by an $N_u \times N$ matrix R in which $R_{u,n}$ is the element at uth row and nth column taking value from $\{0,1\}$. $R_{u,n}=1$ if uth user requests nth file, and 0 otherwise. Users' associations with SBSs are denoted by an $N_u \times N_s$ matrix C in which $C_{u,s}$ is the element at uth row and sth column taking value from $\{0,1\}$. $C_{u,s}=1$ if uth user can be connected with sth SBS, and 0 otherwise.

Therefore, the content placement problem is formulated as follows.

$$\left.\begin{aligned} &\max_{I} \frac{\sum_{u=1}^{N_u}\sum_{n=1}^{N} R_{u,n}\min\left(\sum_{s=1}^{N_s} C_{u,s}I_{s,n},1\right)}{R_{\text{total}}} \\ &\text{s. t. } \sum_{n=1}^{N} I_{s,n} \leqslant M,\ \forall s \end{aligned}\right\} \quad (2)$$

where R_{total} denotes the total number of content requests for all users, and $\min\left(\sum_{s=1}^{N_s} C_{u,s}I_{s,n},1\right)$ is the cache hit indicator when uth user requests nth file. If cache hit happens, it is equal to 1, otherwise 0.

The optimization problem above is a $0-1$ integer programming, which is NP-hard. The main drawback of using the traditional methods to solve the proposed optimization problem is that the complexity is too high and it will increase with the number of edge nodes. Therefore, we propose a heuristic algorithm to approach optimal content placement scheme. The heuristic algorithm proposed in this paper exploits the accuracy of dynamic local popularity model rather than fixed global popularity model, which has been utilized in most existing works. Therefore, the proposed heuristic algorithm can outperform those schemes based on fixed global popularity model via careful clustering of similar SBSs and corresponding local popularity prediction.

2 Local vs global popularity

In this section, we analyze the advantage of employing local popularity over utilizing global popularity in terms of hit probability performance based on Poisson shot noise model which can capture non-stationary popularity.

2.1 Time-varying popularity model

Independent reference model (IRM) has been extensively used for the analysis of caching schemes. However, it assumes that the popularities are fixed which is not appropriate in real world. As a result, Poisson shot noise model (SNM) is proposed[15], which can capture time variation of popularities. Though not necessarily the most accurate model, it can be used for analysis with popularity dynamics.

Each content is modeled as a shot and consists of four characteristics, namely, shape, duration, arrival time and volume. Traverso et al.[15] state that shape and duration are not primary impact factor on the hit performance under online caching policies. Accordingly, rectangular pulses and fixed durations T are adopted for content modeling, thereby facilitating the analysis of optimal caching policy and the according hit performance. The particular analysis in this section needs to be revisited if different content shape and duration model are utilized, but the discussions are adequate to support our point of view on local popularity.

We model the content arrival times as points of a Poisson process with constant rate λ and denote the arrival time of content m by $\bar{t}_m$. The content request volumes are modeled based on a power-law distribution, which is considered to be proper for small time-scale content popularity. More specifically, we denote the request rate of content m by μ_m which is a random variable and constructed as in the following equation.

$$\mu_m = Z_m^{-\alpha}\bar{\mu}(1-\alpha),\ \forall m \quad (3)$$

where Z_m is a uniform random variable in $[0,1]$ and independent for all contents. α represents the power law exponent and $\bar{\mu}$ denotes the mean popularity. Requests for content m are generated as an independent

Poisson process with rate μ_m.

2.2 Hit probability performance with geographical locality of content

Since mobile users generally have diverse sociological and cultural backgrounds and different activities might happen at different sites, it can be expected that content popularities may vary from area to area. For instance, university students are likely to pursue similar contents and hence requests received by nodes on campus regions might have a particular pattern.

1) Geographical locality of content popularity

We denote the local popularity of content m at location l by μ_m^l. The aggregate popularity of content m from all locations is denoted by $\mu_m^{\mathscr{L}}$ and modeled as the model described in Section 3.1. The global mean popularity can be expressed as $\bar{\mu} = E[\mu_m^{\mathscr{L}}]$. We utilize a simple method to model the local cache popularity μ_m^l as follows. Other more accurate models can be used for the analysis in order to capture more complex geographical locality characteristics but are out of the scope of this paper.

$$\mu_m^l = \mu_m^{\mathscr{L}} L \frac{X_m^l}{\sum_{l=1}^{L} X_m^l}, \ \forall m, l \tag{4}$$

where X_m^l is an uniform random variable taking value in [0,1] for each content m at location l.

2) Hit probability performance comparison

Considering geographical locality of content popularities, the popularity distribution observed on a subset of caches is more skewed compared with aggregated popularity over all caches. Higher popularity skewness can take the most advantage of caching technique to improve system performance. Therefore, it can be expected that estimating popularity in clusters can outperform estimating popularity globally in terms of hit performance. We verify this conjecture in theorem. To simplify the analysis, we assume each user can only associate with one SBS at a time and employ caching the most popular policy.

Let $h_{\mathscr{L}}(T)$ be the average hit probability with global popularity $\mu_m^{\mathscr{L}}$, and $h_l(T)$ be the average hit probability at location l with local popularity μ_m^l. Accordingly, the following theorem is derived from the model.

Theorem 1 Considering $T \to \infty$, the average hit probability based on local popularity is higher than that based on global popularity. In other words, for any global popularity distribution $\mu_m^{\mathscr{L}}$, the following inequality holds

$$h_{\mathscr{L}}(\infty) \leqslant h_l(\infty) \tag{5}$$

The equality holds only when local popularities are identical and equal to global popularity.

Proof: Under local popularity, the SBS will cache the locally most popular contents subject to storage limitation. In contrast, under global popularity, each SBS caches the identical most globally popular contents which may not be locally most popular. Since we employ caching the most popular policy and assume that each user can only associate with one SBS at a time, the average hit probability $h_l(\infty)$ at location l equals to the sum of cached locally highest contents' popularities and the average hit probability $h_{\mathscr{L}}(\infty)$ equals to the sum of cached globally highest contents' popularities. Hence, the theorem can be readily proved and we can draw the conclusion that local popularity is more accurate. □

Caching schemes can benefit from the higher accuracy and skewness of the local popularity distribution. Moreover, collecting more samples can make popularity estimation more accurate. Consequently, content popularity aggregation of the subsets of SBSs which receive similar service request could be more efficient. This is discussed in the next section, wherein we firstly identify SBSs with similar local popularity distributions and then predict local popularity from those SBSs only.

3 Content placement scheme

The efficiency of caching policy is predominantly determined by two factors. On one hand, in order to maximize total hit ratio, cached contents should be requested as many times as possible in the future. This raises the question what to cache and need accurate prediction of content requests. On the other hand, if an individual user is able to fetch contents from multiple BSs, caching only most popular contents at each

BS will no longer be the optimal policy. Therefore, where and how to cache is the other question.

Content popularity is generally derived from a remote server as a form of statistical metric during a certain period based on data for enormous users. As a result, the model of request probability is usually assumed as an identical distribution for all users. Zipf-like distribution has been extensively adopted by existing research works as the distribution of content popularity. In other words, the content request of a user is commonly modeled as a random variable with fixed distribution and is identical for all users, that is, independent identical distribution.

As a matter of fact, this independent identical distribution model for content requests can only capture the trend of average user preference. Accordingly, it is more appropriate for cache policy design at server-side. It is well known that humans have personalized preference for contents. This diverse viewing behavior is usually associated with users' demographic information, such as age, gender, occupation and location. Therefore, proactive caching strategy design based on global popularity statistics results in worse performance on total hit rate. Accurate model for content popularity is needed to improve system performance.

Furthermore, people usually stay at fixed places, either home or work. In other words, a user generally gets services from a particular set of SBSs. Therefore, those SBSs belonging to the same area might have similar content popularity. For example, students on campus may have different content preference with employees in an office building, thereby resulting in different content popularity at the SBSs within the campus.

To establish an accurate model of request probability, we facilitate the exploitation of the difference of content popularity between different areas. Accordingly, we first recognize those SBSs belonging to the same area. Then, the request probability for each area is predicted. At last, a content placement policy is designed based on this request probability predicted and edge nodes deployment topology. The detailed analysis and design are given as follows.

All SBSs' download history matrix D over a period of time can be obtained as input measures. And D is an $N_s \times N$ matrix, with each entry $d_{s,n}$ representing the number of download times for nth content by all users through sth SBS.

3.1 SBSs clustering

In order to identify SBSs similar with each other on content requests, we need to find a metric to measure the similarity of SBSs. In recommendation theory, cosine similarity is often used to determine the similarity of users' interest. Since we intend to compare the content popularity of different SBSs rather than the request volume, cosine similarity is a good choice owing to its emphasis on vector direction.

We assume that SBSs with similar download history have similar local content popularity and can be divided into the same cluster. In this paper, we examine cosine similarity between SBSs' download history. Specifically, the similarity $\phi_{i,j}$ between ith and jth SBSs can be calculated as follows

$$\phi_{i,j} = \frac{\sum_f d_{if} d_{jf}}{\sqrt{\sum_f (d_{if})^2}\sqrt{\sum_f (d_{jf})^2}} \qquad (6)$$

where d_{if} denotes the number of download times for content f by all users through SBS i. It is obvious that $\phi_{i,j} \in [0,1]$. The two SBSs tend to have common interests if $\phi_{i,j}$ is closer to 1 than 0. As a result, we can get an $N_s \times N_s$ symmetrical matrix for all SBSs' similarity, with each entry representing similarity between two SBSs.

Each SBS chooses top SBSs with cosine similarity higher than a threshold value γ and forms into a cluster with them. It is obvious that the number of SBSs in a single cluster becomes small with larger threshold value. Furthermore, smaller number of SBSs in each cluster results in more accurate request probability evaluation. The value of the threshold can significantly affect the clustering process as well as the cache hit ratio. In practice, the mobile network operator (MNO) can collect the data of content request history. As a result, they can determine the value of the threshold by analyzing the similarity of different SBSs. As for the study in this paper, we observe that the maximal similarity between two groups of users with different occupations is smaller than 0.98 from the real data sets as in Section 4. Hence, we set the threshold value γ for SBS clustering to 0.98 in our simulation.

3.2 Popularity prediction

In order to identify SBSs similar with each other on content requests, we need to find a metric to measure the similarity of SBSs. In recommendation theory, cosine similarity is often used to determine the similarity of users' interest. Since we intend to compare the content popularity of different SBSs rather than the request volume, cosine similarity is a good choice owing to its emphasis on vector direction.

As Szabo et al.[16] indicate, there exists strong linear correlation between past and future content popularity. Therefore, it is reasonable to use linear regression for the prediction of the request probability. An easy method is to use arithmetic mean. This method, however, overlooks the temporal correlation of content requests. In this work, we use a linear prediction model considering the temporal correlation of content requests. Specifically, we predict the request number $\hat{d}_{\tau f}$ of content f at time slot τ as follows

$$\hat{d}_{\tau f} = \sum_{t=1}^{\tau-1} \theta_{tf} d_{tf} \tag{7}$$

where d_{tf} is the historical request number of content f at time slot t and θ_{tf} is the correspondent parameter.

Determining the parameters is essential to the prediction algorithm performance. In this paper, we aim to find optimal parameters in Eq. (7) in order to achieve lowest prediction accuracy loss. To this end, over-fitting problem is the key challenge to deal with. In particular, some constraints need to be imposed to overcome this problem.

Regularization constraints are proper for generic problems but will overlook the characteristics of our work in hand. In this scenario, content requests are both features and responses of the linear regression. Accordingly, the parameter θ_{tf} can be regarded as the normalized correlation coefficient between the content request of content f at time slot τ and the content request of content f at time slot t. In addition, it is presumed that the correlation coefficients are non-negative, because the content requests in the past can stimulate the content requests in the future. We also assume that shorter time interval means stronger correlation between content requests. Based on above analysis, the optimal parameters can be determined by solving the following optimization problem with linear constraints:

$$\left.\begin{aligned} &\min_{\theta_{tf}} \sum_{t} \sum_{f} (\hat{d}_{tf} - d_{tf})^2 \\ &\text{s.t.}\ \ \theta_{tf} - \theta_{(t+1)f} \geqslant 0, \forall t, f \\ &\qquad \theta_{tf} \geqslant 0,\ \forall t, f \end{aligned}\right\} \tag{8}$$

The optimization problem with linear constraints above can be solved by gradient projection method. And the use of linear constraints for the parameters subsequently brings about fast convergence rate and prevents over-fitting problem.

3.3 Content placement

In this section, we design a content placement policy to approach the upper bound based on the predicted content popularity. Since a user can be connected to multiple SBSs in our model, greedy-caching strategy which caches the most popular contents at each SBS cannot take the advantage of the broadcast nature of radio media. Accordingly, we derive a content placement scheme by virtue of probability theory and convex optimization method. First, we express the total hit ratio in terms of content placement probability and user association probability and formulate the according optimization problem. Then, we prove that the optimization problem is convex and give a sub-optimal content placement solution.

Suppose a user requests file j, and p_j^s denote the probability of the requested file being found at any SBS that the user can associate with. If we can know the probability that the requested file not being found at any SBS, p_j^s equals to 1 minus this probability with the expression as follows:

$$p_j^s = 1 - \sum_{k=0}^{\infty} \alpha_k (1 - c_j)^k \tag{9}$$

where c_j is the probability of a content j being cached in any SBS and α_k is the probability that a user is covered by k SBSs. Then $(1-c_j)^k$ is the probability that none of the covered k SBSs have cached the requested file j.

Given the predicted content popularity and according to our wireless channel model, the expectation of total hit ratio can be expressed as follows:

$$f(c_1,\cdots,c_N) = \sum_{j=1}^{N} \hat{p}_j \left[1 - \sum_{k=0}^{\infty} \alpha_k (1-c_j)^k\right] \tag{10}$$

where $\hat{p}_j$ is the predicted local popularity of file j for a single cluster.

Therefore, the optimal content placement problem can be formulated as Eq. (11) as follows.

$$\left.\begin{aligned} &\max_{c_j} \sum_{j=1}^{N} \hat{p}_j \left[1 - \sum_{k=0}^{\infty} \alpha_k (1-c_j)^k\right] \\ &\text{s. t.} \ \sum_{j=1}^{N} c_j \leqslant M \\ &\qquad c_j \in [0,1],\ \forall j \end{aligned}\right\} \tag{11}$$

It can be readily proved that the objective function in Eq. (11) is concave. Furthermore, the feasible set and constraint functions are all convex. As a result, the problem can be solved as a convex optimization problem. Specifically, the optimal solution can be derive by virtue of Lagrangian relaxation method and Karush-Kuhn-Tucker (KKT) conditions, which is given below as in Eq. (12).

Proposition 1 The optimal primary variables are given with the following expressions,

$$c_j = \begin{cases} 1, & \text{if } \hat{p}_j\alpha_1 > \lambda^* \\ c(\lambda^*), & \text{if } \hat{p}_j\alpha_1 \leqslant \lambda^* \leqslant \hat{p}_j E[k] \\ 0, & \text{if } \hat{p}_j E[k] < \lambda^* \end{cases} \tag{12}$$

where $E[k]$ is the expectation of the random variable k. And λ^* is the optimal dual variable and $c(\lambda^*)$ is the solution of c_j derived from the following Eq. (13).

$$\hat{p}_j \sum_{k=1}^{\infty} \alpha_k k (1-c_j)^{k-1} = \lambda^* \tag{13}$$

and λ^* is the solution of the following equation.

$$\sum_{j=1}^{N} c_j(\lambda^*) = M \tag{14}$$

4 Simulations and performance analysis

To verify our proposed algorithm, we need the data of content download number history at a number of SBSs. As a matter of fact, this kind of reliable comprehensive data of the context information about the request probability is usually private and accordingly difficult to obtain. Nevertheless, we can generate data as realistic as possible by utilizing the public real data sets such as MovieLens[17] for our experiment. The data set of MovieLens is provided by GroupLens and has been widely used in education, research and industry.

Since we intend to investigate the impact of exploiting local popularity rather than global popularity on system performance, demographic information about the users are needed for popularity prediction. To this end, we select the 100 k version of MovieLens data set. The full data sets contain 100 000 ratings by 943 users on 1 682 items and each user has rated at least 20 movies. We consider a movie rating as a request for the content to the wireless network at the particular time point. According to long tail theory, only a small proportion of contents are requested by most subscribers. Therefore, in this simulation, we only use the top 100 movies with the highest request probability. In addition, users' occupation information is also included in the data sets. We assume that the users with the same occupation are within the same area. For instance, students are in school and engineers are in office building.

In the process of the experiment, 20 000 rating records are extracted as the test set, and the others are considered as the training set for learning.

4.1 Evaluation for prediction accuracy

In this section, we evaluate the accuracy performance of the popularity prediction algorithm. We use arithmetic mean prediction as the baseline algorithm. Prediction algorithm with arithmetic mean is formulated as below

$$\hat{d}_{nj} = \frac{\sum_{i=1}^{n-1} d_{ij}}{n-1} \tag{15}$$

As a matter of fact, this arithmetic mean prediction method is linear regression with parameters all equal to $\frac{1}{n-1}$.

To describe the gap between the estimated values and the actual ones, the root mean squared error (RMSE) used in the evaluation criterion is defined as follows:

$$Y = \sqrt{\frac{\sum_{i=1}^{n} | R_i - r_i |^2}{n}} \qquad (16)$$

where R_i and r_i indicate the predicted value and the actual value for the record i, respectively. And n represents the number of rating records in the test set. A smaller RMSE value means better prediction accuracy.

In Fig. 2, we depict the RMSE of both prediction methods as a function of input dimension, which is the number of predictor days we use. Firstly, we are able to observe that the linear regression method outperforms the arithmetic mean method. This result verifies the assumption of linear correlation between past and future content popularity. If they are not correlated, the linear regression method and the arithmetic mean method will be similar in terms of prediction accuracy. Secondly, both methods have the same tendency with the increase of input dimension. Their RMSE both go down at first and then rise. This tendency can be explained as follows. As the dimension increases, the accuracy becomes better because of more input data. However, with the dimension achieves a certain threshold, input data from early days become interference. This might be caused by the fact that users' preference has a periodic pattern. Accordingly, correlation parameter of early days is not the lowest.

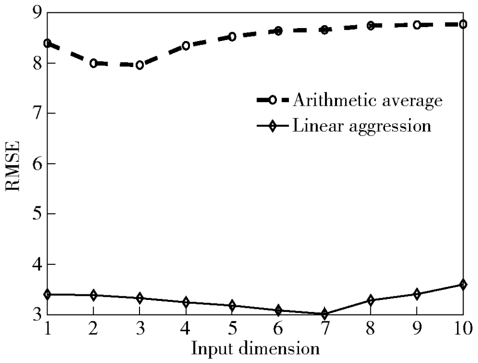

Fig. 2 Prediction accuracy vs input dimension

4. 2 Evaluation for system performance

In this section, we compare the performance of the proposed scheme for content placement which is denoted as clustering based local popularity prediction scheme (CBLPP) with the following benchmark caching schemes:

Random clustering based local popularity prediction scheme (RCBLPP): the SBS clusters are selected randomly other than according to the similarity between SBSs and content placement is designed based on local popularity prediction.

Geographic caching based on global popularity scheme (GCBGP) as inRef. [18]: content placement is designed based on global popularity and probabilistic caching.

Fair caching (FC): each SBS caches the same proportion of every content, which is equivalent to random caching.

Fig. 3 illustrates the total hit ratio of the proposed scheme and baseline schemes with different cache capacity. The total file amount is set to be 100 files and the cache capacity ranges from 1 to 100 files in this figure. Generally, the total hit ratio becomes higher with increase of cache capacity, as more contents can be cached. It can be seen that the proposed scheme outperforms the other three baseline schemes. This is because the proposed scheme captures the request probability more precisely, whereas RCBLPP overlooks the similarity between SBSs and GCBGP neglects the spatial characteristics of content popularity. Note that RCBLPP outperforms GCBGP, which implies that utilizing local content popularity is more sensible than utilizing global content popularity. In addition, the curve of FC scheme is linear and the other three schemes are concave. This is because FC scheme randomly caches contents, thereby more cache capacity just increases the total hit ratio linearly. On the other hand, the other three schemes cache contents according to content popularity and distribution of content popularity is not uniform.

Fig. 4 depicts the total hit ratio of the proposed scheme and baseline schemes with different total file amount. The cache capacity is set to be 20 files and the total file amount ranges from 20 to 100 files in this figure. Overall, the total hit ratio decreases when more contents compared to cache capacity can be requested by users. The proposed scheme outperforms the other three baseline schemes, because it can exploit limited cache capacity more fully by caching the most reques-

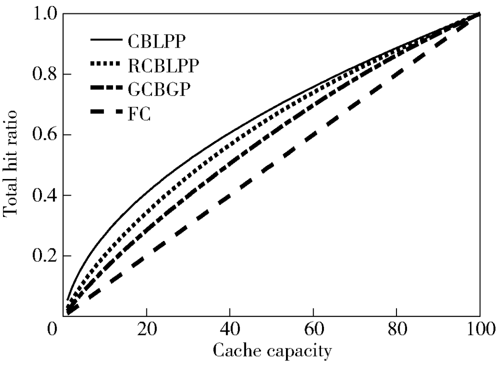

Fig. 3　Total hit ratio vs cache capacity

ted contents. It can be noticed that the decrease of total hit ratio becomes slower with more files. This is because the popularity tail becomes longer with more total files as stated in long tail theory. Furthermore, the curve of FC scheme is not linear as expected in Fig. 5. The reason is that the average popularity for each content does not decline linearly with file quantity rising.

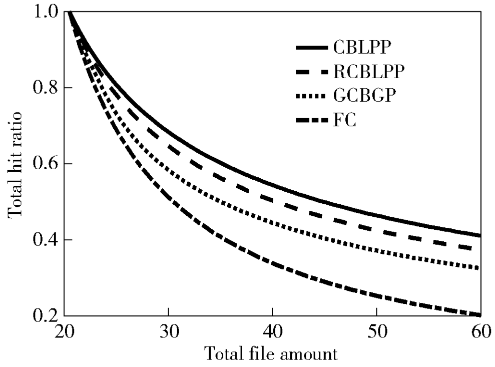

Fig. 4　Total hit ratio vs total file amount

Fig. 5 and Fig. 6 investigate an important system performance metric, namely average transmission delay, of the proposed scheme and baseline schemes with different cache capacity and total file amount, respectively. The total file amount is set to be 100 files and the cache capacity ranges from 1 to 100 files Fig. 5. The cache capacity is set to be 20 files and the total file amount ranges from 20 to 100 files Fig. 6. The average transmission delay is normalized for both simulations and the average transmission rate of SBSs and the MBS is set to be 10 and 1. It can be seen that with limited cache capacity or huge total file amount the average transmission delay can be considerably reduced by employing our proposed caching scheme. The reasoning behind these results is similar with the discussions of Fig. 3 and Fig. 4.

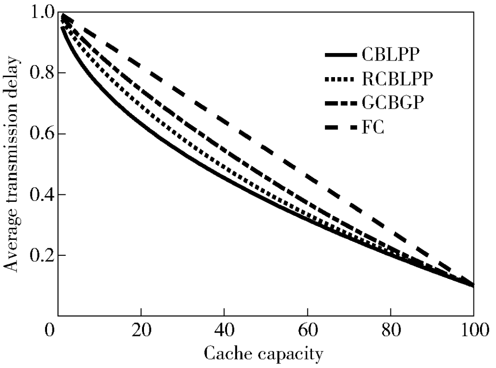

Fig. 5　Average transmission delay vs cache capacity

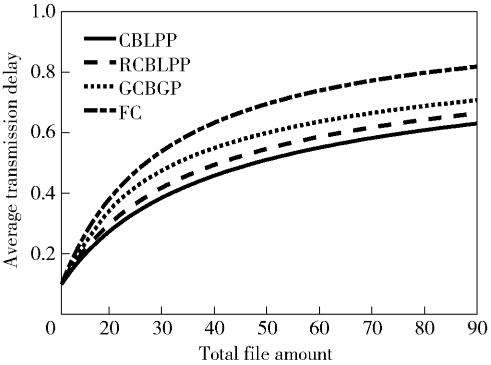

Fig. 6　Average transmission delay vs total file amount

Fig. 7 investigates the influence of different similarity threshold for clustering on total hit ratio with different cache capacity. The total file quantity is set to be 100 files and the cache capacity ranges from 1 to 100 files in this figure. It can be observed that smaller similarity threshold can result in better performance. We can draw an inference from this figure that with smaller similarity threshold for clustering, SBSs which are more similar with each other are grouped together. Accordingly, the popularity for the cluster is more similar with each SBS in the cluster. As a result, the popularity prediction is more precise with lower similarity threshold for each cluster.

Fig. 8 compares the popularity cosine similarity of users with different ages and occupations. Users aging 20 ~ 30, 30 ~ 40, 40 ~ 50 and 50 ~ 60 are compared with users aging 10 ~ 20, respectively. It can be seen that more age difference results in more different taste in contents. In addition, the user preference of lawyer,

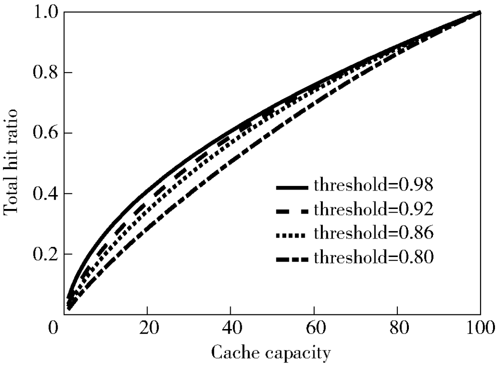

Fig. 7 The influence of different similarity threshold for clustering on system performance

engineer, student and executive are compared with doctors. The implication in this figure is that users with different occupation tend to have different content interest. Hence, clustering SBSs with similar content request probability and utilizing local popularity for different area is rational for unleashing the potential of edge caching technique.

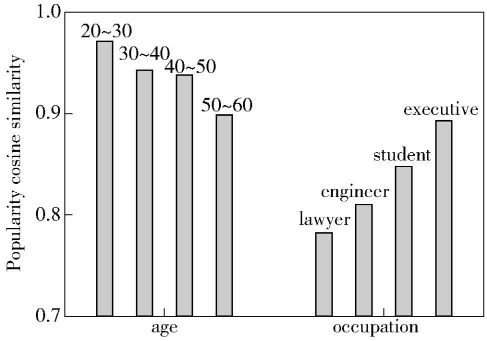

Fig. 8 Popularity cosine similarity in terms of different user age and occupation

5 Conclusions

In this paper, we focus on the problem of content placement with consideration of regional content requests characteristics. A heuristic algorithm is proposed utilizing SBS similarity and probability prediction based on linear regression. Our numerical results show that the proposed scheme can improve system performance in terms of total hit ratio compared to the scheme using global content popularity. In addition, the comparison result of the popularity cosine similarity of users with different ages and occupations imply that content preference observed at different SBSsis diverse.

Despite the new insights provided by our work, some progress can still be made in the future. Since our algorithm is statistical, learning method with better prediction accuracy can be applied to further approach optimal content placement scheme. Beside cache hit ratio, which is considered in this paper, other performance metrics such as network throughput and energy efficiency are also very important and need to be carefully examined. Furthermore, they are usually separately considered in the existing works. Exploring how to balance these different performance metrics is very interesting and beneficial.

References:

[1] Cisco. Cisco visual networking index: global mobile data traffic forecast update, 2016-2021, White Paper[R]. San Jose: Cisco, 2017.

[2] Wang Xiaofei, Chen Min, Taleb T, et al. Cache in the air: exploiting content caching and delivery techniques for 5G systems[J]. IEEE Communications Magazine, 2014, 52(2): 131-139.

[3] Liu Dong, Chen Binqiang, Yang Chenyang, et al. Caching at the wireless edge: design aspects, challenges, and future directions[J]. IEEE Communications Magazine, 2016, 54(9): 22-28.

[4] Maddah-Ali M A, Niesen U. Fundamental limits of caching[J]. IEEE Transactions on Information Theory, 2014, 60(5): 2856-2867.

[5] Wen Juan, Huang Kaibin, Yang Sheng, et al. Cache-enabled heterogeneous cellular networks: optimal tier-level content placement[J]. IEEE Transactions on Wireless Communications, 2017, 16(9): 5939-5952.

[6] Liu Dong, Yang Chenyang. Caching policy toward maximal success probability and area spectral efficiency of cache-enabled HetNets[J]. IEEE Transactions on Communications, 2017, 65(6): 2699-2714.

[7] Chen Min, Hao Yixue, Hu Long, et al. Green and mobility-aware caching in 5G networks[J]. IEEE Transactions on Wireless Communications, 2017, 16 (12): 8347-8361.

[8] Kiskani M K, Sadjadpour H R. Throughput analysis of decentralized coded content caching in cellular networks [J]. IEEE Transactions on Wireless Communications, 2017, 16(1): 663-672.

[9] Liu Tingting, Li Jun, Shu Feng, et al. Design of contract-based trading mechanism for a small-cell caching

system[J]. IEEE Transactions on Wireless Communications, 2017, 16(10): 6602-6617.

[10] Shi Fang, Tan Weiqiang, Xia Junjuan, et al. Hybrid content placement for physical-layer security in cooperative networks[J]. IEEE Access, 2018, 6: 8098-8108.

[11] Chen Youjia, Ding Ming, Li Jun, et al. Probabilistic small-cell caching: performance analysis and optimization[J]. IEEE Transactions on Vehicular Technology, 2017, 66(5): 4341-4354.

[12] Zeng Ming, Lin T H, Chen Min, et al. Temporal-spatial mobile application usage understanding and popularity prediction for edge caching[J]. IEEE Wireless Communications, 2018, 25(3): 36-42.

[13] Wang Yanfeng, Ding Mingyang, Chen Zhiyong, et al. Caching placement with recommendation systems for cache-enabled mobile social networks[J]. IEEE Communications Letters, 2017, 21(10): 2266-2269.

[14] He Shuo, Tian Hui, Lyu Xinchen. Edge popularity prediction based on social-driven propagation dynamics[J]. IEEE Communications Letters, 2017, 21(5): 1027-1030.

[15] Traverso S, Ahmed M, Garetto M, et al. Temporal locality in today's content caching: why it matters and how to model it[J]. ACM SIGCOMM Computer Communication Review, 2013, 43(5): 6-12.

[16] Szabo G, Huberman B A. Predicting the popularity of online content[J]. Communications of the ACM, 2010, 53(8): 80-88.

[17] Harper F M, Konstan J A. The MovieLens datasets: history and context[J]. ACM Transactions on Interactive Intelligent Systems (TiiS), 2016, 5(4): 1-19.

[18] Blaszczyszyn B, Giovanidis A. Optimal geographic caching in cellular networks[C]//2015 IEEE International Conference on Communications (ICC). New York: IEEE Press, 2015: 15437268.