Performance Trade-off of Wireless Coded Cache

Sinong Wang, Hao Feng, Xiaohua Tian, Hui Liu and Xinibin Wang
Department of Electronic Engineering, Shanghai Jiao Tong University
{snwang, fenghao, xtian, huiliu, xwang8}@sjtu.edu.cn

***Abstract*—Recent studies show that the multicast and cache can facilitate the wireless content distribution by mitigating the wireless traffic rate during the peak-traffic time. These techniques are based on the assumption that many requested contents are initiated in the peak-traffic time. Instead, the user requests are asynchronous and these techniques actually implement one extreme case of the trade-off between the traffic and delay. To illustrate the main idea, we first construct the information-theoretical lower bound and find the lowest traffic is inverse proportional to the delay. Then, we investigate two schemes under two scenarios in the single bottleneck network. Explicitly, the first scheme strives for minimum complexity by resorting to the simple cache procedure and pure multicast procedure; the second scheme is based on the famous coded cache scheme that consists of a cooperative cache procedure and coded multicast procedure. Besides, we investigate the effect of the slotted transmission and show this technique can provide a constant gain. Compared to the constructed lower bounds, the proposed schemes are both shown to achieve the order optimal trade-off between traffic and delay. The coded cache scheme under delay constraint scenario has the same order with pure multicast scheme. As far as we know, this is the first work to address this problem.**

I. Introduction

The wireless data traffic volume beyond 2020 is predicted to be 1000 times higher than that in 2010 [1], which will place tremendous pressure in modern infrastructure such as backhaul and base station. This explosive tendency is driven mainly by the proliferation of smart user terminals (UT) and various multimedia request.

One promising way is to bring the content closer to user terminals via caching in order to reduce data traffic volume to accomodate such an explosive increase. The wireless network presents a high temporal variability in network traffic volume since the network resources are extremely scarce in certain time periods and comparatively idle in others. Then some caching techniques are proposed to balance the traffic load over the wireless link, in a word, popular contents are partially prefetched into user terminals during the off-peak time so the number of contents needed to be delivered in the peak-traffic time can be reduced, with requesters experiencing less congestions [2]– [5].

Another main point to alleviate such pressure is to multicast the contents to group of users. This method is based on the observation that the multimedia contents are not consumed with the same frequency, at least statistically [6]. In fact, among the huge quantity of requests, the users access a small portion of most popular contents and always send the same request. The base station can multicast a content to group of users who have the same request. Then the multiple requests can be satisfied by only one transmission, thus reduce the traffic over the wireless.

Based on above two techniques, a novel scheme called coded cache is recently proposed in the single bottleneck network [7], which exploits the cooperation of different user caches and the coded multicast to significantly reduce the network traffic. The major feature distinguishes the coded caching from its uncoded counterpart is that the data delivered over the wireless link are coded data over multiple contents, so that multiple requests for even different contents can be satisfied with a single coded transmission. They theoretically prove the required delivery-phase traffic volume produced by this scheme only has a constant gap compared to information-theoretical lower bound.

However, this technique requires the base station to collect requests from every users, which leads to a large waiting time or delay for pre-arrival users. Thus, it should be exploited under some delay constraint. In fact, if we adopt the multicasting and caching technique in such network, the gain mostly derives from the number of users the base station gathers and the number of requests being merged or coded, and these parameters are dependent of the user delay. Thus, there exists a traffic-delay trade-off in such network. For example, in one extreme case, if we restrict the maximum tolerant delay is zero, the base station will lose ability to gather user requests and only unicast content to each user, which produces the largest traffic. In another extreme case, if the max tolerant delay is large enough to collect all users' requests in a cellular network, the base station can take a smart multicast scheme considering the characteristics of each user's local cache and their requests, thus the traffic can be minimized.

There are large amount of works investigating the trade-off between delay and throughput in Device-to-Device(D2D) network and ingenious results have been presented [12]– [14]. The way in which delay scales for traffic optimal schemes in caching network, however, has not been well-studied. Indeed, it is unclear what "delay" and "traffic" precisely means, especially in such settings. One of our work is the definition of "delay" and "traffic", which is both meaningful and theoretical tractable. As far as we know, this is the first work this addresses the problem and our main contributions are three-fold:

- We point out the traffic of the single bottleneck network can be reduced by sacrificing users' delay.
- We investigate the trade-off between user's delay and traffic in such network under two kinds of user demands and construct the information theoretical lower bound of the traffic under given delay.
- Explicitly, we present three kinds of schemes with progressively increasing complexity and analyze their delay-traffic trade-off under the given conditions. Based on the

constructed lower bound, we prove these schemes are order optimal and only exhibits multiplicative gap.

The remainder of the paper is organized as follows. The preliminary is spread in Section II. Next, we present the service model and main results. In Section IV and V, we first construct the information theoretical lower bound and investigate the trade-off for three schemes under two scenarios. Numerical results are given in Section VI. Finally, we summarize this work and present the future work in the last section.

II. Preliminary

In this section, we introduce the recently proposed scheme, called coded cache, which exploits the cooperation of different user caches and the coded multicast to significantly reduce the network traffic. Besides that, we briefly introduce the user request model to facilitate such trade-off analysis in the sequel.

A. Coded Cache

In the coded caching scheme, contents are divided into segments and partially prefetched during the off-peak-traffic time, and content retrieval requests issued during the peak-traffic time are responded by multicasting coded data over different contents. Moreover, an efficient caching scheme is proposed in [8], in which the content placement is performed in a decentralized manner by developing a caching algorithm that creates simultaneous coded-multicasting opportunities without coordination in the placement phase. While in [7], [8], it is assumed that the contents popularity is uniformly distributed. When considering the nonuniformity of content popularity distribution, some more coded caching schemes are developed. Knowing the popularity distribution of the contents in the base station, they classify contents into several groups and execute the coded caching for each group independently [9]. We make use of the coded caching scheme from [8] in this paper. Therefore, we now briefly overview this scheme.

The set in [8] assumes that the popularity of each content is uniform. It considers a similar architecture that has N contents and K users. Each user has a local cache of size MF bits with $M \leq N$. The codec scheme under such scenario operates as follows. In the placement phase, each user randomly caches the same amount MF/N bits of each content. Then in the later delivery phase, all users send their requests and the server transmits multiple coded signals $\oplus_{k\in U} V_{k,U/\{k\}}$[1] over the shared link to users to retrieve their requested contents. This scheme achieves a peak traffic rate of

$$K \cdot (1-\frac{M}{N}) \cdot \frac{N}{KM} \cdot (1-(1-M/N)^K)F, \tag{1}$$

which is shown to be within a constant multiplicative gap of the information theoretical lower bound. Illustration of this scheme is provided in Example 1.

Example 1 (Codec scheme in [8]). Suppose that there is a simple system distributing $N=2$ contents A and B to $K=2$ users, each with the cache size MF bits. In the placement phase, each user randomly caches $MF/2$ bits of content A and B independently. Let us focus on content A. The operations of placement phase partitions content A into four subcontents, $A=(A_\varnothing, A_1, A_2, A_{1,2})$. A_U denotes the bits of content A that are prefetched in the memories of users in U, where $U \subset \{1,2\}$. For example, A_1 represents the bits of A only available in first user's memory. We adopt $|\cdot|$ as the operation of size, thus, $|A_\varnothing| = (1-M/2)^2F$ bits, $|A_1| = |A_2| = (M/2)(1-M/2)F$ bits, $|A_{1,2}| = (M/2)^2F$ bits. The same analysis holds for content B.

In the delivery phase, we assume that user 1 and user 2 request content A and B, respectively. User 1 has accessed to subcontent A_1 and $A_{1,2}$ in its local cache and lacks $A_\varnothing$ and A_2. Similarly, user 2 has already accessed to B_2 and $B_{1,2}$, and lacks $B_\varnothing$ and B_1. In traditional uncoded caching scheme, the server is required to unicast $A_\varnothing$ and A_2 to user 1 and unicast $B_\varnothing$ and B_1 to user 2. The total rate is

$$2 \cdot \frac{M}{2}(1-\frac{M}{2})F + 2(\frac{M}{2})^2F = MF \text{ bits},$$

Under the coded caching manner, the server can satisfy the requests by transmitting $A_\varnothing$, $B_\varnothing$ and $A_2 \oplus B_1$ over the shared link, where $\oplus$ denotes the bit-wise XOR operation. The rate over the shared link is

$$\frac{M}{2}(1-\frac{M}{2})F + 2(\frac{M}{2})^2F = \frac{MF}{2} + \frac{M^2F}{4} < MF \text{ bits},$$

For other possible user requests, this rate is also achievable.

From **Example 1**, it can be found the gain of uncoded cache scheme comes from the isolated cache memory, called the *local cache gain*, captured by a factor $(1-M/N)$ in (1), which linearly decreases with the cache size MF. Instead, the main gain of codec scheme comes from the multicasting opportunities created by a jointly designed placement and delivery phase, which can be captured by a factor called *global cache gain*, $N/K/M \cdot (1-(1-M/N)^K)$ in (1). This gain is inverse proportional to the cache size M and much more significant in aspects of reducing the traffic volume.

B. User Request model

Requests from users arrive at the base station according to a random process. Typically we model the arrival process as Poisson process. It is a viable model when the requests from a large population of independent users. Under the right circumstances, the number of requests from users arriving during a fixed time interval is a random variable obeying the Poisson distribution.

In probability theory and statistics, the Poisson distribution is a discrete probability distribution that expresses the probability of a given number of events occurring in a fixed interval of time and space if these events occur with a known average rate and independently of the time since the last event [15].A discrete random variable X is said to have a poisson distribution if the random variable X only takes non-negative integer $0, 1, 2, ...$, and the probability mass function of X is given by:

$$\mathbb{P}(X=k) = \frac{(\lambda T)^k e^{-\lambda T}}{k!} \tag{2}$$

[1] $V_{k,U/\{k\}}$ denotes the bits of the content d_k requested by user k cached exclusively at users in U

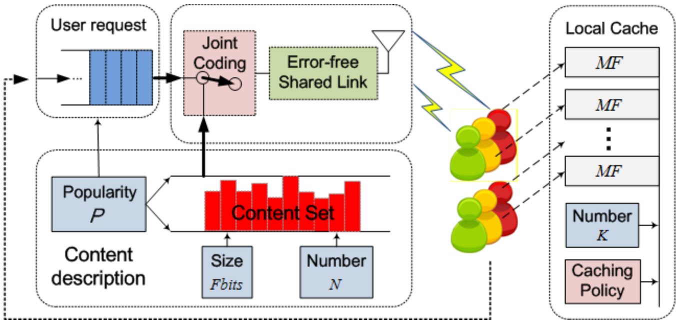


Fig. 1: System model

where λ is a constant parameter, independent of the time t, and independent of arrivals in earlier intervals. λ is called the arrival rate and λ. T is the time period.

III. System Model and Main Results

A. System Model

We consider a single bottleneck network where a base station serves some users via a shared wireless link, each user has a local cache of size MF bits. There are N contents in the base station, each has the same size of F bits. The contents requested by the different users from the base station are independent of each other and obeys a content popularity distribution $P = [p_1, p_2, \ldots, p_N]$, where p_i is the probability that a user requests content i.

In practical scenario, the channel state of each user is fluctuated and different among them, thus the number of bits carried in each channel might be different. There are many resource allocation methods to counteract this fading effect, such as the power control and bandwidth or time slot allocation. This line of work, however, is not pursued here and we only rely on the scheme how to cache and how to make a delivery in the application level. Thus, we assume the contents are distributed through an error-free shared link to users such as in the long term evolution (LTE) broadcasting system [16], where the error-free can be achieved with error correction scheme or reliable transmission scheme in the upper layer.

The system operates in two phases. In the placement phase, part of contents are prefetched in each user's local cache. As the placement phase happens in the off-peak time such as the late night, the wireless traffic incurred in this phase is tolerable. In the delivery phase, any user k sends its request d_k in the time t_k and follows a poission arrival model (2) with rate λ. The base station collects amount of requests in its buffer without exceeding delay constraint, then it delivers specific data through the shared wireless link to those users. Each user recover its requested contents using both the local cached data and the data just received over wireless.

Based on above system model, we formulate the following traffic-delay trade-off problem:

Definition 1. ***(Traffic-delay trade-off Problem)*** *For given delay constraint and the arrival rate of user requests, how to harness the peak-time traffic in the delivery phase?*

Intuitively, increasing the delay will reduce the traffic, since the base station can collect more user requests and operate smartly, such as multicasting transmission, joint encoding and priority encoding transmission. Thus, this problem is such a trade-off problem between traffic and delay. We first define a partition of a set, then define these two parameters based on partition.

Definition 2. ***(Partition of set)*** *For a given sequence set $S = \{t_1, t_2, \ldots, t_L\}$ and $N_L = \{n_1, n_2, \ldots, n_L\}$ with $L \to \infty$, where t_i is the arrival time of ith request and k_i is the content id of ith request. We define a partition $N_L = \{U_1, U_2, \ldots, U_h\}$ of set N_L that satisfies*

$$\bigcup_{1 \le i \le h} U_i = N_L,$$

$$U_i \bigcap U_j, \forall i, j, i \neq j, 1 \le i, j \le h.$$

Remark that the partition of the set divide the user request sequence into many groups and U_i can be regarded as the a request vector for ith transmission. Based on this concept, we provide the definition of the delay and traffic.

Definition 3. ***(Delay)*** *The delay D of our model is defined as follows. There exists a partition $U = \{U_1, U_2, \ldots, U_h\}$ of set N_L that satisfies*

$$E_{U_i}\left[\max_{k \in U_i} t_k - \min_{k \in U_i} t_k\right] \le D. \tag{3}$$

A brief comment on the notion of delay adopted in our work is now in order. This kind of delay represents the average maximum tolerant delay between the instant representing the start of a request of a user, and the instant representing the start of transmission of a packet that can be tolerated for every delivery. It means that, in each transmission of the base station, the user's experienced delay can be larger or less than D. The only constraint is that the long-term average delay is less than D, which can be guaranteed by the specific strategy. Note this notion can be regarded as the ergodic trade-off analysis instead of the outage trade-off analysis.

In fact, D refers to the queuing delay in the network layer and does not consider the other kinds of delay such as processing delay, propagation delay and transmission delay. Since the queuing delay in our system dominates the main part, and as argued in the sequel, this delay analysis offers a lower bound on the total delay that any transmission should obey this delay constraint.

Definition 4. ***(Traffic)*** *The traffic of a scheme Γ is defined as the average traffic volume produced in the shared link to satisfy one request,*

$$R = \lim_{L \to \infty} \frac{1}{L} \sum_{i=1}^{h} R_v(U_i, \Gamma). \tag{4}$$

Where $R_v(M, U_i, \Gamma)$ is the traffic of each delivery under specific scheme Γ and requested vector U_i.

Remark that this traffic is an average traffic for each request, with the unit of F bits.

B. Main Results and Outline

We investigate the trade-off between traffic and delay under two scenarios: uniform and nonuniform content popularity distribution. Under each scenario, we first construct the information-theoretical lower bound of the traffic under given delay constraint; then, we design the schemes, including the caching and the delivery strategy, to approximate this bound. In the uniform case, the popularity of each content is identical and there exists some trival schemes that are order optimal. In the second scheme, the popularity of each content is different and the design of such order optimal scheme is a non-trival procedure, especially the design of the caching strategy, i.e., how to allocate the cache space for each contents in accord with the content popularity distribution.

Under uniform user demands, we show a possible lower bound of this system is $R=\Theta\left(\frac{1}{D}\right)$. Then, we propose three classes of schemes with progressively increasing complexity to approximate this lower bound. Both of them scale as the same law of $R=\Theta\left(\frac{1}{D}\right)$ while producing different multiplicative gap. Furthermore, the following technical assumption is imposed.

We let

$$D=\max\left\{\omega\left(\left(1-\left(1-\frac{1}{N-M}\right)^{1-\frac{M}{N}}\right)^{-1}\right),\omega\left(\frac{N}{M}\right)\right\}. \tag{5}$$

This technical assumption is made to ensure (as shown in the sequel) that the average traffic is not donminated by the scaling behaviour of delay. Note that for constant N, M and K, this assumption is always established.

1) The first scheme strives for minimum complexity by resorting to a simple caching rule that all users store the same contents, along with non-coded multicast and yields a scaling law of
$$R=\frac{N-M}{\lambda}\Theta\left(\frac{1}{D}\right). \tag{6}$$
2) The second, and more complex, scheme is based on the famous coded cache technique, benefiting from the cooperation between the cache phase and coded multicast, yielding the scaling law of
$$R=\frac{N-M}{\lambda M}\Theta\left(\frac{1}{D}\right). \tag{7}$$
This scheme shows a constant gain M over the first scheme.
3) Moreover, we introduce the concept of the slotted transmission, which exploits the transmission delay of each request and only delivers these contents in each time slot. This concept can further improve the number of requests gathered in each transmission, thus further reduce the traffic. We incorporate this concept into above schemes and show that, under small delay, the gain is a piece-wise constant, while under large delay, they will degrade to the unslotted transmission and no gain can be obtained.

Under nonuniform user demands, that means, some contents are hot and always being required while others are desolate and rarely being required. In our analysis, we assume $p_1\leq p_2\leq\cdots\leq p_N$ and introduce a caching distribution $Q=[q_1,q_2,\ldots,q_N]$ to describe the specific caching strategy, where q_i denotes the proportion of content i occupying each user's cache. Furthermore, the following technical assumption is imposed. We let

$$D=\max\left\{\max_{1\leq i\leq N}\left\{\frac{1}{p_i}\right\},\max_{M+1\leq i\leq N}\left\{\left(1-(1-p'_i)^{1-S_M}\right)^{-1}\right\}\right\}, \tag{8}$$

where $p'_i=p_i\left(\sum_{j=M+1}^{N}p_j\right)^{-1}$ and $S_M=\sum_{i=M+1}^{N}p_i$. This technical assumption is also made to ensure (as shown in the sequel) that the average traffic is not donminated by the scaling behaviour of delay.

According to the lower bound constructed under the uniform user demands, we show a possible lower bound of nonuniform demands is also $R=\Theta\left(\frac{1}{D}\right)$ under condition (8).

1) The first scheme takes a same non-coded multicast delivery and we prove the optimal caching distribution is
$$q_i=\frac{1}{M}, M+1\leq i\leq N; 0, 1\leq i\leq M. \tag{9}$$
Based on this caching distribution, the pure multicast scheme yields a same scaling law of (6).
2) The second scheme is much more complicated. It also takes a coded multicast method of coded cache technique. Then, we determine an order optimal caching distribution,
$$q_i^{\dagger}=\frac{1}{NM}\left[1+\frac{1}{\lambda D}\ln\left(\frac{p_i^N}{\prod\limits_{j=1}^{N}p_j}\right)\right], 1\leq i\leq N. \tag{10}$$
This caching distribution yields the scaling law,
$$R=\frac{(N-1)\left(1-\left(\prod\limits_{i=1}^{N}p_i\right)^{\frac{1}{N}}\right)}{\lambda}O\left(\frac{1}{D}\right). \tag{11}$$

IV. Uniform User Demand Scenario

In this section, we consider the uniform user demand scenario where the content has the same popularity that $p_1=p_2=\cdots=p_N$. Under this scenario, we can take the fair cache space allocation strategy, i.e., each user cache the same M contents or each user's local cache is divided equally to N parts and stores different parts of these N contents.

A. Lower Bound

The lower bound of the traffic under given delay constraint is independent of any practical schemes. We first transform the lower bound of R into the lower bound of the traffic under fixed number of requests, then construct the lower bound of traffic under fixed number of requests based on the concept of cut-set bound.

Lemma 1. *The lower bound of the traffic is,*

$$R_{lb}^{e}(D)=\sum_{k=0}^{\infty}\frac{(\lambda D)^{k-1}e^{-\lambda D}}{k!}\overline{R_{lb}^{e}}(k). \tag{12}$$

where $\overline{R^e_{lb}}(k)$ denotes the lower bound of the traffic under k requests.

Proof: According to the definition of traffic,

$$R^e_{lb}(D) = \lim_{L\to\infty} \frac{1}{L} \sum_{i=1}^{h} R^{lb}_v(U_i). \tag{13}$$

where $R^{lb}_v(U_i)$ is the lower bound of traffic under requested vector U_i. Since the user request behaviour can be regarded as a a wide-sense stationary process, thus we take the statistical average to represent above long sample of the process,

$$\begin{aligned} R^e_{lb}(D) &= \frac{1}{\lambda D} E\left[R^{lb}_v(U_i)\right] \\ &= \frac{1}{\lambda D} \sum_{k,U_i} R^{lb}_v(U_i) \mathbb{P}\left[|U_i| = k, U_i \in U\right] \\ &= \frac{1}{\lambda D} \sum_{k,U_i} R^{lb}_v(U_i) \mathbb{P}\left[|U_i| = k\right] \mathbb{P}\left[U_i \in U \big| |U_i| = k\right]. \end{aligned}$$

Note that

$$\begin{aligned} \mathbb{P}\left[|U_i| = k\right] &= \frac{(\lambda D)^k e^{-\lambda D}}{k!}, \\ \mathbb{P}\left[U_i \in U \big| |U_i| = k\right] &= \frac{1}{N^k}. \end{aligned}$$

Hence,

$$R^e_{lb}(D) = \sum_{k=0}^{\infty} \frac{(\lambda D)^{k-1} e^{-\lambda D}}{k!} \sum_{|U_i| = k} \frac{R^{lb}_u(U_i)}{N^k}. \tag{14}$$

According to the definition of the $\overline{R^e_{lb}}(k)$,

$$R^e_{lb}(D) = \sum_{k=0}^{\infty} \frac{(\lambda D)^{k-1} e^{-\lambda D}}{k!} \overline{R^e_{lb}}(k). \tag{15}$$

■

Remark this lemma shows that we can calculate the traffic based on the number of requests, instead of the specific identifier of the requests. The main reason is that the definition of the traffic and delay.

Theorem 1. *For given N contents, delay D. The user requests arrive at a rate of λ and each user has a cache of size $0 \leq M \leq N$,, the lower bound of the traffic is,*

$$R^e_{lb}(D) = \Omega\left(\frac{1}{D}\right). \tag{16}$$

Proof: We use the results of [7] that is based on a cut-set bound argument, we can get

$$\overline{R^e_{lb}}(k) \geq \max_{s \in \{1,\ldots,\min\{N,k\}\}} \left(s - \frac{s}{\lfloor N/s \rfloor} M\right). \tag{17}$$

Further, we can get

$$\overline{R^e_{lb}}(k) \geq \max_{s \in \{1,\ldots,\min\{N,k\}\}} s\left(1 - \frac{sM}{N-s}\right). \tag{18}$$

We choose $s = cM/N$ that $s \in \{1, \ldots, \min\{N, k\}\}$ and then,

$$\overline{R^e_{lb}}(k) \geq \frac{cM - c^2 - c^2 M}{M - c} \cdot \frac{N}{M}. \tag{19}$$

Then, based on the lemma 1, we can get,

$$R^e_{lb}(D) \geq \frac{cM - c^2 - c^2 M}{M - c} \cdot \frac{N}{M} \cdot \frac{1}{\lambda D}. \tag{20}$$

Hence,

$$R^e_{lb}(D) = \Omega\left(\frac{1}{D}\right). \tag{21}$$

■

This information theoretical lower bound shows that, in the bottleneck network, sacrificing delay can reduce the traffic over the wireless and the traffic is at most inverse proportional to the delay. In general, a key insight from the procedure of above proof is that, if the traffic under N contents and k requests is bounded by the number of contents N instead of the number of requests k, it will show an trade-off $\Theta\left(\frac{1}{D}\right)$. For example, if we take the traditional unicast scheme, the traffic under N contents and k requests is $k(1 - M/N)$ and shows a trade-off $R = 1 - M/N$ that is independent of delay D. Instead, if we take the multicast transmission, the same requests can be satisfied by one transmission and will be bounded by the number of contents. Based on this insight, we consider the following two schemes to approximate this bound.

B. Scheme with Pure Multicast: Γ^e_u

In this class of scheme Γ^e_u, we take a simple cache rule that each user prefetches the same M contents in the placement phase. Since the contents have a uniform popularity distribution, we choose from first M contents. Then, in the delivery phase, the base station collects users' requests under the delay constraint. From later analysis, we can find that, even each user prefetches the different contents, it will also show the same trade-off, since there is no coding among different requests and the content can be regarded as equal. Then, the base station multicasts the requested contents to group of users that has the same request, respectively. This procedure does not consider the coded opportunities among different requests and refers to pure multicast. The details of this scheme can be seen in the Algorithm 1.

Algorithm 1: Pure multicast scheme with uniform demands: Γ^e_u

```
Placement Phase
for ($k = 0; k < K; k++$) do
    User $k$ prefetches the first $M$ contents ;
Delivery Phase
At $k$th delivery, the base station collects users' requests
without exceeding the maximum tolerant delay $D_k$, that
$D_k$ satisfies $\sum_{i=1}^{k} D_k \leq kD$;
for ($i = 0, i < G; i++$) do
    Multicast the ith requested content to the users in the
    ith group ;
```

Remark the delay constraint in the Algorithm 1 is a possible implementation of our definition of delay. It guarantees the average delay between current delivery and previous deliveries is less than D, thus it can guarantee the long-term average delay is less than D.

Theorem 2. *Under the condition (5), the average traffic produced by scheme Γ_u^e scales as*

$$R^e(D, \Gamma_u^e) = \frac{N-M}{\lambda} \Theta\left(\frac{1}{D}\right). \tag{22}$$

Proof: see Appendix A. ■

Theorem 2 provides the trade-off produced by scheme Γ_u^e. It can be found that the trade-off produced by this scheme has the same order of the lower bound and has a constant gain that is linear to the cache size M.

C. Scheme with coded Multicast: Γ_c^e

This scheme is based on the decentralized coded cache scheme in [8] with considering the delay constraint. The main procedure is same to the scheme Γ_c^e in Section II. We now focus on the trade off analysis of this scheme.

Algorithm 2: Coded multicast scheme with uniform demands: Γ_c^e

```
Placement Phase
for $(k = 0; k < K; k++)$ do
    for $(n = 0; n < N; n++)$ do
        user $k$ randomly prefetches $MF/N$ bits of
        content $n$ ;
Delivery Phase
At $k$th delivery, the base station collects users' requests
without exceeding the maximum tolerant delay $D_k$, that
$D_k$ satisfies $\sum_{i=1}^{k} D_k \leq kD$;
for $(k = K, k > 0; k--)$ do
    for choose $k$ users from $K$ users to form a subset $U$
    do
        server sends $\oplus_{k \in U} V_{k,U/\{k\}}$ to users in $U$.
```

Theorem 3. *Under the condition (5), the average traffic produced by scheme Γ_c^e scales as*

$$R^e(D, \Gamma_c^e) = \frac{N-M}{\lambda M} \Theta\left(\frac{1}{D}\right). \tag{23}$$

Proof: see Appendix B. ■

It can be found that the scheme Γ_c^e also produces a same order delay gain compared to scheme Γ_u^e, while it provides a larger constant gain of $1/M$. This constant gain comes from the cooperative of the distributed cache and coded multicast, which can guarantee the different requests can be satisfied by one transmission.

D. Scheme with slotted-transmission

In this section, we present a concept that can further reduce the constant gap of above schemes. This concept exploits the transmission delay of each content. Above schemes assume that in the delivery phase all contents are transmitted instantaneously. Factually, there exists the transmission delay for each content in the shared link. We consider a time-slotted transmission and assume each content requires J time slots T_c to deliver and the total transmission delay is JT_c. During the transmission of some contents, other previously inactive users might submit their requests to the server, then we consider following steps: if these current requests can wait until the next time slot without violating delay constraint, we operate the multicast scheme between the current requests and the contents being transmitted in the next time slot, else, we unicast these requested contents. The key insight in this concept is that it utilizes the transmission delay to gather the user requests, which not only increases the number of requests in one transmission, but also maintains the original delay. The following example shows the explicit procedure.

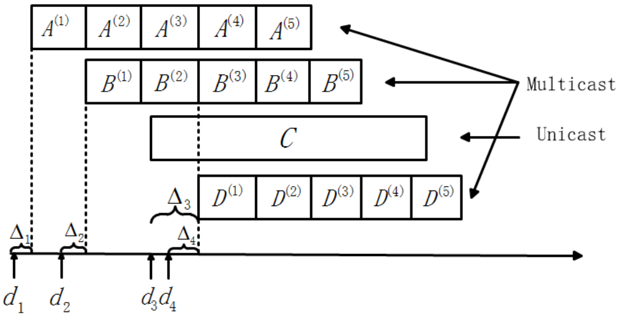


Fig. 2: The example about slotted transmission.

Example 2. Consider a system with $N = 4$ contents A, B, C and D, and $K = 4$. We first divide each contents into J consecutive subcontents, i.e., $A = (A^{(1)}, A^{(2)}, \ldots, A^{(J)})$, such that each time slot can serves each subcontent. Specifically, we choose $J = 5$ in this example.

In the placement phase, we simply regard each segment as a distinct content and apply the cache rule in Algorithm 1. For the delivery phase, consider the first request $d_1 = A$. The server responds by unicasting of the first part $A^{(1)}$ of content A. Meanwhile, previously inactive user submit request $d_2 = C$, as shown in Fig. 2. The request d_2 is laid aside until completing the delivery of $A^{(1)}$. Then it starts the transmission of $A^{(2)}$ and $C^{(1)}$ in the next time slot and use the non-coded or coded multicast scheme, the rest part of content A and C is handled similarly. Assume that in later time, another two requests $d_3 = B, d_4 = D$ are revealed. If we react to request d_3 in the next time slot, the delay will exceed D. Thus, we unicast this requested content B and the server reacts to request d_4 in next time slot by delivering $A^{(4)}, C^{(3)}$ and $D^{(1)}$, as shown in Fig. 2. The procedure continues in the same manner as above steps.

We find that the request d_2, d_3 d_4 experience delays of Δ_2, 0 and Δ_4. Since this scheme responds each user's request in the next time slot, the max delay experienced by each user is less than T_c. Further, if we responds each user's request every two or more time slots, the delay experienced by each user is less that $2T_c$ or more, while it can gather more users to participate in the coded mulitcast and reduce more traffic.

We incorporate this concept into the scheme Γ_u^e and Γ_c^e, and refer the new schemes as Γ_{us}^e and Γ_{cs}^e, respectively. The following theorems show their trade-offs.

Theorem 4. *If D satisfies $(\rho-1)T_c \leq D \leq \rho T_c$ and $1 \leq \rho \leq J$, the average traffic $R^e(D, \Gamma_{us}^e)$ produced by scheme Γ_{us}^e is*

$$\frac{\rho(N-M)}{D}\left(\frac{1-e^{\frac{J\lambda D}{\rho}\left[\left(1-\frac{1}{N-M}\right)^{1-\frac{M}{N}}-1\right]}}{J\lambda M}+\frac{\rho T_c - D}{N}\right), \tag{24}$$

else, it scales as

$$\frac{N-M}{\lambda M}\Theta\left(\frac{1}{D}\right). \tag{25}$$

Proof: See in Appendix C. ■

Theorem 5. *If D satisfies $(\rho-1)T_c \leq D \leq \rho T_c$ and $1 \leq \rho \leq J$, the average traffic produced by scheme Γ_{cs}^e is*

$$R^{es}(D, \Gamma_{cs}^e) = \frac{\rho(N-M)}{D}\left(\frac{1-e^{-\frac{J\lambda DM}{\rho N}}}{J\lambda M}+\frac{\rho T_c - D}{N}\right), \tag{26}$$

else, the average traffic scales as

$$R^{es}(D, \Gamma_{cs}^e) = \frac{N-M}{\lambda M}\Theta\left(\frac{1}{D}\right). \tag{27}$$

Proof: See Appendix D ■

According to the Theorem 4 and 5, the trade-off between traffic and delay under slotted transmission has a piecewise form. Under small delay, it has a compound form of inverse proportionality and negative linearity. The main reason is that this scheme consists of two parts: unicast and coded multicast, and they have different effects on the traffic under given D. While under large delay, it has a same form of previous schemes, since the transmission delay JT_c is much too smaller compared to D and the transmission process can be regarded as instantaneous. Besides, from the process of proof, we can find that the effect of the slotted transmission is transforming the original poisson stream of user requests into another poisson stream that has a higher arrival rate $J\lambda D/\rho$. Thus, it can gather more user requests and further reduce the traffic.

Corollary 1. *If D satisfies $(\rho-1)T_c \leq D \leq \rho T_c$ and $1 \leq \rho \leq J$, the traffic provided by scheme Γ_{us}^e and Γ_{cs}^e have constant gain $J/\rho, 1 \leq \rho \leq J$ over scheme Γ_u^e and Γ_c^e when N is large enough and T_c is small enough, else, it has no gain.*

Proof: See Appendix E. ■

From Corollary 1, we can find that the gain is piece wise and become smaller when ρ increases. Considering the complexity of the system is linear to the number of contents JN, this result provides a possible guideline how to choose a appropriate J. The system predefines a delay D, then we can determine the ρ and J to maximize such gain.

V. Nonuniform User Demand Scenario

When the contents have different popularity such as $p_1 \leq p_2 \leq \cdots p_N$, it is inappropriate to take above fair cache rule. A tendentious caching strategy is needed, i.e., allocate more space to the more popular content. For analytical convenience, we introduce the caching distribution Q to describe such a tendentious caching strategy.

A. Lower bound

The information theoretical lower bound is independent of any delivery scheme and any caching distribution. Theorem 6 presents a possible lower bound based on average cut-set bound argument.

Theorem 6. *For given N contents, content popularity distribution, delay D and condition (8). The user requests arrive at a rate of λ and each user has a cache of size $0 \leq M \leq N$,*

$$R_{lb}^d(D) = \Omega\left(\frac{1}{D}\right). \tag{28}$$

Proof: See Appendix F. ■

The lower bound under nonuniform content popularity distribution is same as the uniform case, since we impose the condition (8) to elimate the effect of the content popularity distribution. In fact, if we consider this effect, above lower bound will be less, while it will become mathematically intractable.

B. Scheme with Pure Multicast

This scheme does not consider the coded opportunities in the delivery phase and is same as delivery procedure of the scheme Γ_u^e, the only difference is the caching strategy in the placement phase. It requires each user prefetching the same $q_i MF$ bits of content $i, 1 \leq i \leq N$. Note that different caching distribution will produce a different traffic and might drive a different trade-off between delay and traffic. For comparison among different schemes, we determine an optimal caching distribution to minimize the traffic and analyze the trade-off behaviour under such optimal caching distribution.

OPT:

$$\text{Min} \quad R^d(D, Q, \Gamma_u^d) = \sum_{k=0}^{\infty} \frac{(\lambda D)^{k-1} e^{-\lambda D}}{k!} \overline{R^d}(k, Q, \Gamma_u^d) \tag{29}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} q_i = 1, 1 \leq q_i \leq \frac{1}{M}, \tag{30}$$

where the constraint is to avoid violating caching capacity constraints and caching duplicated packets for each user.

$\overline{R^d}(k, Q, \Gamma_u^d)$ is the average traffic under scheme Γ_u^d, k requests and caching distribution Q. According to the definition, we can get

$$\overline{R^d}(k, Q, \Gamma_u^d) = \sum_{i=1}^{N} \left[1 - (1-p_i)^k\right](1 - q_i M) \tag{31}$$

Thus, based on (31), we can transform above **OPT** model into the following model.

$$\text{Min} \quad \sum_{i=1}^{N} e^{-\lambda D p_i} \cdot q_i \tag{32}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} q_i = 1, 1 \leq q_i \leq \frac{1}{M}, \tag{33}$$

Then, the optimal caching distribution is given by

$$Q_u^* = \arg\max_{Q} \sum_{i=1}^{N} e^{-\lambda D p_i} \cdot q_i. \tag{34}$$

Lemma 2. *The optimal caching distribution under scheme Γ_u^d is*

$$q_i = \frac{1}{M}, M+1 \le i \le N; 0, 1 \le i \le M. \tag{35}$$

Proof: Since the content popularity satisfies $p_1 \le p_2 \le \dots p_N$, then $e^{-\lambda D p_1} \ge e^{-\lambda D p_2} \ge \dots e^{-\lambda D p_N}$. We divide the content sets into two subsets: $L = \{1, 2, \dots, M\}$ and $H = \{M+1, M+2, \dots, N\}$. Assume a caching distribution $Q^{'} : q_i^{'} = 1/M, i \in H; q_i^{'} = 0, i \in L$ that only caches the contents in popular content set H. We choose $j \in L, k \in H$ and let $q_j = \alpha, q_k = 1/M - \alpha, \alpha > 0$ in $Q^{'}$ to produce another caching distribution $Q^{''}$ that caches some contents in less popular content set. Then, we compare the objective function under these two caching distributions,

$$\sum_{i=1}^{N} e^{-\lambda D p_i} \cdot \left(q_i^{'} - q_i^{''}\right) = \alpha\left(e^{-\lambda D p_k} - e^{-\lambda D p_j}\right) < 0.$$

Thus, cache the contents in the less popular set will increase the objective function, which means that the caching distribution $Q^{'}$ is optimal. ■

According to the Lemma 2, the optimal caching strategy for the pure multicast scheme is prefetching the M most popular contents. The following Theorem shows the trade-off produced by this caching strategy.

Theorem 7. *Under the condition (8) and caching distribution Q_u^*, the traffic produced by scheme Γ_u^d scales as*

$$R^d(D, Q_u^*, \Gamma_u^d) = \frac{N-M}{\lambda}\Theta\left(\frac{1}{D}\right). \tag{36}$$

Proof: See Appendix G. ■

Theorem 7 shows that the trade-off of pure multicast scheme in nonuniform case equals to the trade-off in the uniform case under specific condition.

C. Scheme with coded Multicast

In the scheme Γ_c^d, we take the same delivery procedure of scheme Γ_c^e. Then, we consider the following caching strategy. It requires each user randomly prefetching $q_i MF$ bits of content $i, 1 \le i \le N$. Then we take the same procedure to derive the optimal caching distribution for scheme Γ_c^d. Due to the internal combinatory structure of the coded cache scheme, it is much more difficult to determine a close-form expression of the optimal caching distribution. We first construct an original model to formulate this problem as the nonlinear programming model. Then, we exploit the characteristics of the coded cache to develop a concept-relax model that falls into the category of the convex optimization. Based on the generalized Lagrangian multiplier method, we derive an order optimal caching distribution.

OPT:

$$\text{Min} \quad R^d(D, Q, \Gamma_c^d) = \sum_{k=0}^{\infty} \frac{(\lambda D)^{k-1} e^{-\lambda D}}{k!} \overline{R^d}(k, Q, \Gamma_c^d), \tag{37}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} q_i = 1, 1 \le q_i \le \frac{1}{M}, \tag{38}$$

where $\overline{R^d}(k, Q, \Gamma_c^d)$ is the average traffic under scheme Γ_c^d, k requests and caching distribution Q.The following lemma shows the way to determine $\overline{R^d}(k, Q, \Gamma_c^d)$.

Lemma 3. *Under the request situation $\vec{s} = (\alpha_1, \alpha_2, \dots, \alpha_N)$ and the given content popularity distribution P, where α_i denotes the number of users requesting content i and satisfies $\alpha_1 + \alpha_2 + \dots + \alpha_N = k$, then,*

$$\overline{R^d}(k, Q, \Gamma_c^d) = \sum_{\vec{s}} \mathbb{P}(\vec{s}) \cdot R_{\vec{s}}(Q, \Gamma_c^d), \tag{39}$$

where $R_{\vec{s}}(Q, \Gamma_c^d)$ denotes the average traffic under request situation $\vec{s}$, caching distribution Q and specific scheme Γ_c^d, and calculated by

$$\sum_{i=1}^{k} \sum_{v \subset [k], |v| = i} \max_{j \in v} \{(q_{d_j} M)^{i-1} (1 - q_{d_j} M)^{k-i+1}\}. \tag{40}$$

Proof: See Appendix K. ■

It shows the traffic is only related to the number of requests for each content, and independent of who requests which content. The proof of Lemma 3 is based on a symmetric analysis.

$\mathbb{P}(\vec{s})$ is the the probability of request situation $\vec{s}$ occurring. Consider that α_1 users request content 1, α_2 users request content 2, α_3 users request content 3, ..., and α_N users request content N, the number of all possible cases for such an event is $C_K^{\alpha_1} \cdot C_{K-\alpha_1}^{\alpha_2} \cdot C_{K-\alpha_1-\alpha_2}^{\alpha_3} \cdots C_{K-\alpha_1-\dots-\alpha_{N-1}}^{\alpha_N}$, and the probability of these cases are identical, thus,

$$\mathbb{P}\left(U_i \in U \big| |U_i| = k\right) = \frac{k!}{\alpha_1! \cdot \alpha_2! \cdot \dots \cdot \alpha_N!} \cdot \prod_{i=1}^{N} p_i^{\alpha_i}. \tag{41}$$

Then the optimal caching distribution is given by

$$Q_c^* = \arg\min_{Q} R^d(D, Q, \Gamma_c^d). \tag{42}$$

We now consider the computation complexity of the optimization problem OPT. The computation complexity of calculating $R_{\vec{s}}(Q, \Gamma_c^d)$ is $O(2^k)$ and the number of all possible $\vec{s}$ is equivalent to the number of solutions of equation: $\sum_{i=1}^{N} \alpha_i = k$, which is C_{k+N-1}^{N-1}. Thus, the computation complexity of obtaining optimal caching distribution Q^* under $R^d(D, Q, \Gamma_c^d)$ is at least $O(2^k \cdot C_{k+N-1}^{N-1})$. It can be found that the complexity increases exponentially as N, k and obtaining the optimal caching distribution is unavailable when the number of users and contents become large. Nevertheless, it provides a feasible approach to compute the optimal solution as a benchmark.

Since the computational complexity of the problem OPT is high, it is natural to ask if there is any equivalence, which can obtain a solution that is approximate to the optimal one but incurs low computational complexity. We note that the traffic over wireless is used to deliver the part of content that has not been cached on the mobile devices. In an extreme case that every content has been cached locally, there is no need of the wireless traffic in the delivery phase; therefore, minimizing the average size of uprefetched contents can reduce the traffic volume. The corresponding problem formulation is as follows.

OPT-Relax:

$$\text{Min} \quad \sum_{k=0}^{\infty} \frac{(\lambda D)^{k-1} e^{-\lambda D}}{k!} V(k, Q, \Gamma_c^d) \tag{43}$$

$$\text{s.t.} \quad \sum_{i=1}^{N} q_i = 1, 1 \leq q_i \leq \frac{1}{M}, \tag{44}$$

where $V(k, Q, \Gamma_c^d)$ is the average size of uprefetched contents under k users, and can be calculated by

$$V(k, Q, \Gamma_c^d) = \sum_{i=1}^{N} p_i (1 - q_i M)^k. \tag{45}$$

Then, the objective function is transformed into

$$\sum_{i=1}^{N} p_i e^{\lambda D M q_i}. \tag{46}$$

Theorem 8. *The optimal caching distribution $Q^\dagger$ under the **OPT-Relax** model is*

$$q_i^\dagger = \frac{1}{NM} \left[1 + \frac{1}{\lambda D} \ln \left(\frac{p_i^N}{\prod\limits_{j=1}^{N} p_j} \right) \right], 1 \leq i \leq N. \tag{47}$$

Proof: See in the Appendix I. ■

This theorem provides a relaxed solution to the original model. It seems like a kind of luffing method: for those contents with higher popularity, up in the average line $\frac{1}{NM}$; for those contents with lower popularity, down in the average line $\frac{1}{NM}$. The proof is based on a convex analysis and general Lagrangian multiplier method.

We take the caching distribution $Q^\dagger$ as the approximate caching distribution for the **OPT** model. Due to the intractable form of $R^d(D, Q, \Gamma_c^d)$, we first construct upper bound of it, which is much more facilitate to analyze, then we show the trade-off between traffic and delay under caching distribution $Q^\dagger$ and upper bound.

Lemma 4. *For N contents, requests arrival rate λ and delay D. If $p_1 \leq p_2 \leq \cdots p_N$ and $q_1 \leq q_2 \leq \cdots q_N$, then,*

$$R^d(D, Q, \Gamma_c^d) \leq \frac{1}{\lambda D} \sum_{i=1}^{N} \mathbb{P}(A_i) \frac{1 - q_i M}{q_i M} \left(1 - e^{-\lambda D M q_i}\right) \tag{48}$$

equal if only if $p_1 = p_2 = \cdots = p_N$ and $q_1 = q_2 = \cdots = q_N$. Where $\mathbb{P}(A_i)$ represents the probability that k users request content $i, i+1, \ldots, N$, and can be calculated as

$$\mathbb{P}(A_i) = (1 - \sum_{j=1}^{i-1} p_j)^k \cdot [1 - (1 - \frac{p_i}{1 - \sum_{j=1}^{i-1} p_j})^K]. \tag{49}$$

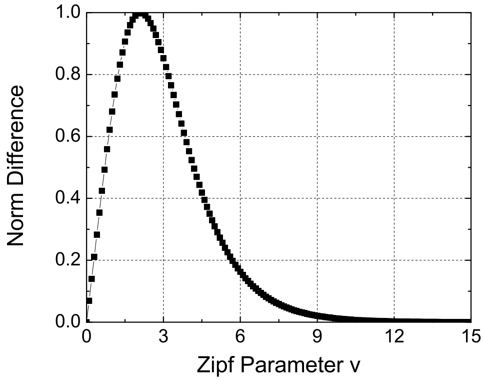


Fig. 3: The normalized difference between $R^d(D, Q, \Gamma_c^d)$ and its upper bound with respect to the maximal difference, with $N = 3$, $D = 1$ and $\lambda = 3$.

Proof: See Appendix J ■

Lemma 4 gives the upper bound of incurred wireless traffic volume given P, Q and scheme Γ_c^d in the general case. In order to investigate the tightness of it, we apply the Zipf distribution [6] that has been widely adopted in modeling media content popularity, caching distribution $Q^\dagger$ derived from our relaxed model and the scheme Γ_c^d, and we can find the normalized difference between the upper bound traffic and the yielded traffic with respect to the maximal difference with $N = 3$, $D = 1$ and $\lambda = 3$ as shown in Fig. 3. The Zipf parameter v describes how concentrate the content popularity is. With v increases, the popularity concentration level of contents increases, the relaxation error incurred by reducing the caching distributions for contents $i+1, i+2, ..., N$ to q_i in deriving the upper bound increases thus the difference increases as shown in Fig. 3. After Zipf parameter v achieves to a certain level, the users are just interested in a few most popular contents, which makes the contents with smaller indices unlikely to be requested thus decrease the relaxation error. That is why Fig. 3 has a shape of mountain-top.

Theorem 9. *Under the condition (8), the traffic produced by scheme Γ_c^d and caching distribution $Q^\dagger$scales as*

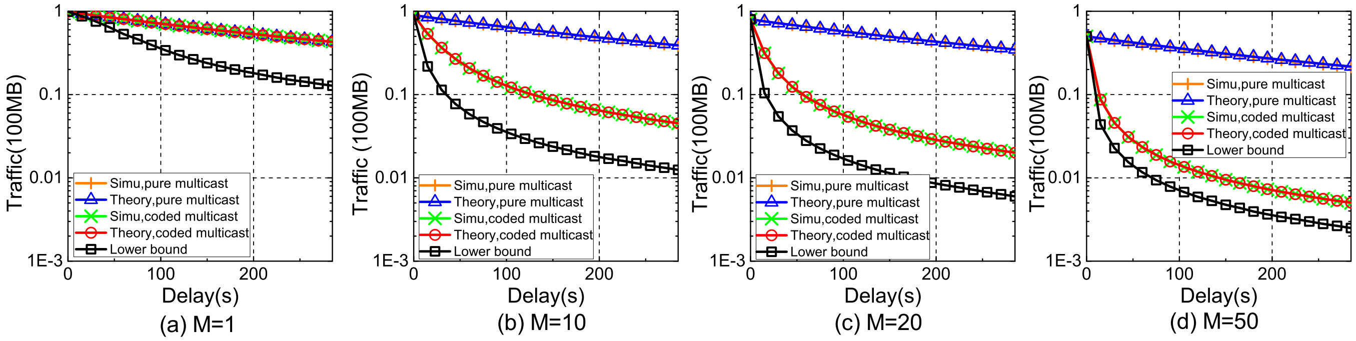
$$R^d(D, Q^\dagger, \Gamma_c^d) = \frac{(N-1)\left(1 - \left(\prod\limits_{i=1}^{N} p_i\right)^{\frac{1}{N}}\right)}{\lambda} O\left(\frac{1}{D}\right). \tag{50}$$

Proof: See Appendix K. ■

Theorem 9 shows that the traffic produced by scheme Γ_c^d is $O\left(\frac{1}{D}\right)$.

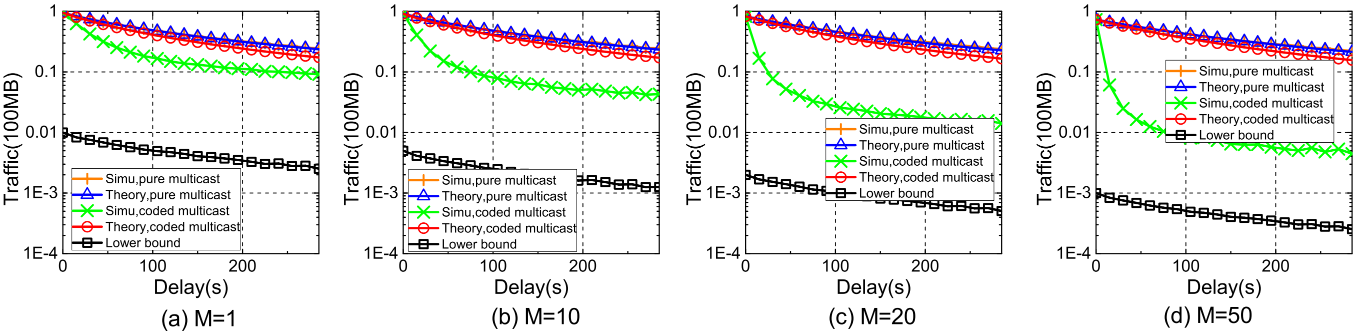
VI. Numerical Results

In this section, we validate the theoretical analysis by providing numerical results on the network performance and compare them to the analytical results. We develop a simulation platform based on the system model on Matlab to emulate behaviour of such network. In this platform, we consider a circular cell with no inter-cell interference from other cells, which has a radius of 600m and users are randomly and independently distributed in the cell. The content is of size 100MB, which is reasonable by assuming a screen size 1280×720 and 150 seconds playing time. Based on the platform, intensive



(a) M=1 (b) M=10 (c) M=20 (d) M=50

Fig. 4: The theoretical and simulation trade-off produced by different schemes under uniform user demands.



(a) M=1 (b) M=10 (c) M=20 (d) M=50

Fig. 5: The theoretical and simulation trade-off produced by different schemes when $v = 0.6$.

simulations are performed, from which we obtain statistics about trade-off between traffic and delay. We first evaluate the accuracy of the derived analytical expressions. The impact of content popularity distribution and network parameters is then discussed as following.

A. *Uniform user demands*

The theoretical results is based on the accurate description (54) and (57) instead of the scaling law, since the scaling law can be attained straightforward by these equations. The lower bound is based on equation (19). As shown in Fig. 4, the theoretical results are in excellent agreement with the simulation results and they all show a inverse proportional fashion respect to the delay. The coded cache scheme outperforms the pure multicast scheme under small delay, while shows a constant gap under large delay.

We further investigate how the cache size affects above schemes. From Fig. (4)(a) to Fig. (4)(d), we operate these two schemes under different cache size $M = \{1, 10, 20, 50\}$ and observe that, when the cache size increases, the gain of coded cache scheme increases much faster than the pure muticast scheme and approximates to the theoretical lower bound. The main reason is that the coded cache scheme exploits the cooperation of distributed cache and introduces a *global cache gain* or a *coded gain*, which is higher order than the traditional pure multicast scheme [8].

A fundamental problem in this setting is how much traffic saving can be obtained via sacrificing the delay. For example, when the cache size is small, as shown in Fig. (4)(b), we can see that when the delay increases from 1 to 10, the traffic is reduced about 3.4% and 30% under pure multicast scheme and coded cache scheme, respectively. When the cache size is large, as shown in Fig. (4)(b), we can see that when the delay increases from 100 to 200, the traffic is reduced about 30% and 60% under pure multicast scheme and coded cache scheme, respectively. Note this traffic reduce effect is strengthen when the cache size increases.

B. *Nonuniform user demands*

We adopt the Zipf distribution [6] to describe the nonuniform user demands.

$$p_i = \frac{\frac{1}{i^v}}{\sum\limits_{j=1}^{N} \frac{1}{j^v}}, 1 \le i \le N. \tag{51}$$

The Zipf parameter v describe the skewness of the distribution that the distribution shows large skewness under large v and vice versa. In most cases, the Zipf parameter $v = 0.6$ and we fix it in the following simulation. The theoretical results of pure multicast scheme is also based on the accurate description (77) and the theoretical results of coded cache scheme is based on its upperbound (94). Note that the different cache distribution will affect the trade-off of the coded cache scheme. We first adopt the caching distribution $Q^{\dagger}$ for comparison. Then we investigate how the cache distribution affect such trade-off.

As shown in Fig. 5, we compare above two schemes under different cache size $M = \{1, 2, 5, 10\}$. The theoretical results of pure multicast scheme is in excellent agreement with its simulation results, since we take a accurate description of trade-off. While the theoretical results of coded multicast scheme is much larger than the simulation results, since the theoretical results are based on the upper bound. The simulation results of coded multicast scheme approximates to the theoretical lower bound and shows a significant gain of decreasing traffic when delay increases. Compared to the uniform case, the traffic decreasing gain is more obvious in the nonuniform cases. As shown in Fig. (5)(b), we can see that when the delay increases from 1 to 10, the traffic is reduced about 10% and 50% under pure multicast scheme and coded cache scheme, respectively. Besides that, when the cache size increases, the pure multicast scheme shows little gain while the coded multicast scheme shows a significant gain of decreasing the traffic.

Further, we investigate how the cache distribution affect such trade-off for the coded multicast scheme. Here we compare the $Q^{\dagger}$ with the well-known Least Recently Used (LRU) caching scheme and the baseline scheme in [11].

LFU: In such a scheme, each user prefetches the M most popular contents in its local cache.

Baseline Scheme: This scheme divides the contents set into two groups and fairly prefetches the contents of the first group.

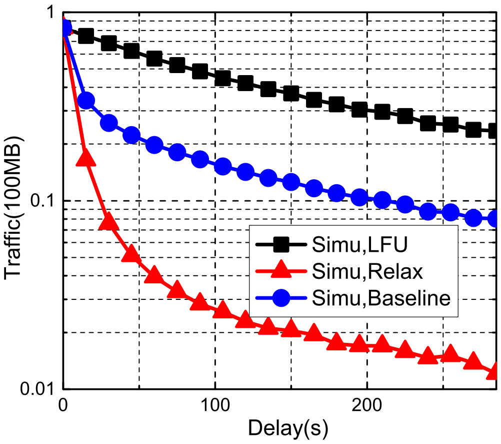


Fig. 6: The theoretical and simulation trade-off produced by different schemes when $v = 0.6$.

Fig. 6 shows the traffic versus the delay under different caching distribution. It can be found that $Q^{\dagger}$ produces the lowest traffic and the baseline scheme is laid between the $Q^{\dagger}$ and LFU. The trade-off produced by $Q^{\dagger}$ shows apparent inverse proportional manner while another two caching distributions produces the partial inverse proportional manner. The main reason behind this trend is that LFU and baseline scheme only prefetches part of contents, which decreases the coded opportunities among different requests, thus diminishes such global cache gain in the transmission.

VII. Conclusion and Future Work

This paper has addressed the *traffic-delay trade-off* problem in the single bottleneck network and has presented multiple schemes to implement it, including pure multicast and code multicast scheme. We theoretically show the trade-off produced by both schemes under two kinds of scenarios: uniform user demands and nonuniform user demands. For the second scheme, the optimized caching distribution have been derived with the content popularity distribution taken into account. Based on the constructed lower bound, both schemes show the order optimality with respect to the delay. Numerical results have shown that the theoretical analysis are in excellent agreement with the simulation results. Further, we have found the so called *global cache gain* produced by coded cache scheme has the same order with the delay, which factually provided another dimension to reduce the traffic. For example, for the delay-intense scenario, we can deploy large cache size to the user terminals, while for the storage-intense scenario, we can set larger delay constraint.

Our future work is focusing on analyzing such trade-off when the delay is independent of the system parameters, and investigating the effect of such parameters to show if there exists an order gain. Another direction is to design the more efficient scheme to implement such trade-off.

References

[1] Cisco, "Cisco visual networking index: Global mobile data traffic forecast update 2012-2017," *Whiter paper*, 2013.

[2] N. Golrezaei, A. F. Molisch, A. G. Dimakis, G. Caire, "Femtocaching and device-to-device collaboration: A new architecture for wireless video distribution," *IEEE Communications Magazine*, vol. 51, no. 4, pp.142–149, 2013

[3] J. Llorca, A. M. Tulino, K. Guan, J. Esteban, M. Varvello, N. Choi, and D. Kilper, "Network-coded caching-aided multicast for efficient content delivery," in *Proc. IEEE ICC*, 2013, pp.3557–3562.

[4] J. Llorca and A. M. Tulino, "The content distribution problem and its complexity classification," *Alcatel-Lucent technical report*, 2013.

[5] S. Gitzenis, G. S. Paschos, L. Tassiulas, "Asymptotic laws for joint content replication and delivery in wireless networks," *IEEE Transactions on Information Theory*, vol. 59, no. 5, pp. 2760–2776, 2013

[6] M. Cha, H. Kwak, P. Rodriguez, Y. Ahn, and S. Moon, "I tube, you tube, everybody tubes: Analyzing the worlds largest user generated content video system," in *Proc. ACM SIGCOMM*, 2007, pp. 1–14.

[7] M. A. Maddah-Ali, U. Niesen, "Fundamental limits of caching," *IEEE Transactions on Information Theory*, vol. 60, no.5, pp. 2856–2867, 2014.

[8] M. A. Maddah-Ali, Niesen U, "Decentralized coded caching attains order-optimal memory-rate trade-off," *IEEE/ACM Transactions on Networking*, vol. PP, no.1, pp. 1, 2014.

[9] U. Niesen and M. A. Maddah-Ali, "Coded caching with nonuniform demands," in *Proc. IEEE INFOCOM*, 2014, pp. 221–226.

[10] M. Ji, A. M. Tulino, J. Llorca, G. Caire, "Order optimal coded caching-aided multicast under Zipf demand distributions," Feb. 19, 2014, Online: http://arxiv.org/abs/1402.4576.

[11] M. Ji, G. Caire and A. F. Molisch, "Order optimal coded caching-aided multicast under Zipf demand distributions," in *Proc. IEEE Information Theory Workshop*, 2013, pp. 1–5.

[12] A. E. Gamal, J. Mammen, B. Prabhakar, S. Devavrat, "Throughput-Delay Trade-off in Wireless Networks," in *Proc. IEEE INFOCOM*, 2004, pp. 464–475.

[13] X. Lin and N. B. Shroff, "Towards Achieving the Maximum Capacity in Large Mobile Wireless Networks," in *IEEE Journal of Communications and Networks*, vol. 6, no. 4, pp. 352-361, 2004.

[14] M. J. Neely, E. Modiano, "Capacity and delay trade-offs for ad hoc mobile networks," in *IEEE Transactions on Information Theory*, vol. 52, no.6, pp. 1917–1937, 2005.

[15] R. D. Yates, D. J. Goodman, "Probability and stochastic processes," *John Wiley and Sons*, 1999.

[16] EXPWAY, Online: http://www.expway.com/.

[17] D. D. Sleator and R. E. Tarjan, "Amortized efficiency of list update and paging rules," *ACM Communications*, vol. 28, no.2, pp. 202–208, 1985.

[18] M. Chrobak and J. Noga, "LRU is better than FIFO," in *Proc. ACM SODA*, 1999, pp. 78–81.

[19] R. Pedarsani, M. A. Maddah-Ali and U. Niesen "Online coded caching,"Nov. 14, 2013, Online: http://arxiv.org/abs/1311.3646.

[20] G. S. Ladde and D. D. Siljak, "Convergence and stability of distributed stochastic iterative processes," *IEEE Transactions on Automatic Control*, vol. 35, no. 6, pp. 665–672, 1990.

[21] G. S. Ladde and M. Sambandham, "Random difference inequalities," *North-Holland Mathematics Studies*, vol. 110, pp. 231–240, 1985.

[22] A. E. Raftery "A model for high-order Markov chains," *Journal of the Royal Statistical Society*, vol. 47, no. 3, pp. 528–539, 1985.

[23] G. Szabo and B. A. Huberman, "redicting the popularity of online content," *ACM Communications*, vol. 53, no. 8, pp. 80–88, 2010.

[24] M. Cha, H. Kwak, P. Rodriguez, Y-Y. Ahn and S. Moon, "I tube, You Tube, Everybody Tubes: Analyzing the Worlds Largest UserGenerated Content Video System," in *Proc. ACM SIGCOMM conference on Internet measurement*, 2007, pp. 1–14.

[25] X. Cheng, J. Liu and C. Dale, "Understanding the characteristics of internet short video sharing: A YouTube-based measurement study," *IEEE Transactions on Multimedia*, vol. 15, no. 5, pp. 1184–1194, 2013.

[26] V. Chandrasekhar, J. G. Andrews, and A. Gatherer, "Femtocell networks: a survey," *IEEE Communications Magazine*, vol. 46, no. 9, pp. 59–67, 2008.

[27] http://traces.cs.umass.edu/index.php/Network/Network.

Appendix A
Proof of Theorem 2

We take a same procedure in the proof of Theorem 1, we can get,

$$R^e(D,\Gamma_u^e)=\sum_{k=0}^{\infty}\frac{(\lambda D)^{k-1}e^{-\lambda D}}{k!}\overline{R^e}(k,\Gamma_u). \tag{52}$$

The $\overline{R^e}(k,\Gamma_u^e)$ denotes the average traffic under scheme Γ_u^e and k requests, which can be calculated by the following way.

Since each user prefetches first M contents, the average number of users should be transmitted by the share link is actually $k(1-M/N)$. We define $(N-M)$ binary random variable $X_i=0,1,M+1\leq i\leq N$, which equals to 1 if and only if the content i is requested by at least one user. Then,

$$\begin{aligned}\overline{R^e}(k,\Gamma_u)&=E\left[\sum_{i=M+1}^{N}X_i\right]=\sum_{i=M+1}^{N}E\left[X_i\right]\\&=(N-M)\left(1-\left(1-\frac{1}{N-M}\right)^{k\left(1-\frac{M}{N}\right)}\right).\end{aligned} \tag{53}$$

Thus,

$$R^e(D,\Gamma_u^e)=\frac{N-M}{\lambda D}\left\{1-e^{\lambda D\left[\left(1-\frac{1}{N-M}\right)^{1-\frac{M}{N}}-1\right]}\right\}. \tag{54}$$

Under the condition (5), we can get

$$R^e(D,\Gamma_u^e)=\frac{N-M}{\lambda}\Theta\left(\frac{1}{D}\right). \tag{55}$$

Hence it is clear that the assumption on D in (5) ensures that the average traffic is not dominated by the scaling behaviour of D.

Appendix B
Proof of Theorem 3

We take a same procedure in the proof of Theorem 2, we can get

$$R^e(D,\Gamma_c^e)=\sum_{k=0}^{\infty}\frac{(\lambda D)^{k-1}e^{-\lambda D}}{k!}\overline{R}(k,\Gamma_c^e). \tag{56}$$

Based on the results of [8], we can get

$$\overline{R^e}(k,\Gamma_c^e)=\frac{N-M}{M}\left[1-\left(1-\frac{M}{N}\right)^k\right]. \tag{57}$$

Hence,

$$R^e(D,\Gamma_c^e)=\frac{N-M}{\lambda M}\Theta\left(\frac{1}{D}\right). \tag{58}$$

Appendix C
The proof Theorem 4

In the time slot i, there are k_i requests, including current requests and previous uncompleted transmitted requests, should be served in the multicast way. Thus, the traffic produced by the multicast way is

$$\begin{aligned}R_m^e(D,\Gamma_{us}^e)&=\lim_{T\to\infty}\frac{J}{TE[k_i]}\sum_{i=1}^{T}\frac{\overline{R}(k_i,\Gamma_{us}^e)}{J}\\&=\frac{E_{k_i}\left[\overline{R}(k_i,\Gamma_{us}^e)\right]}{E[k_i]}\\&=\frac{\sum\limits_{k=0}^{\infty}\mathbb{P}(k_i=k)\overline{R}(k_i,\Gamma_{us}^e)}{E[k_i]}\end{aligned}$$

When the delay D falls into the range of $(\rho-1)T_c\leq D\leq\rho T_c$, the average number of requests that will be participated in every transmission is $E[k_i]=J\lambda D/\rho$. Thus,

$$\mathbb{P}\left(k_i=k\right)=\frac{(J\lambda D/\rho)^k e^{-J\lambda D/\rho}}{k!} \tag{59}$$

Based on the results of (54), we can get

$$R_m^e(D,\Gamma_{us}^e)=\frac{\rho(N-M)}{J\lambda MD}\left\{1-e^{\frac{J\lambda D}{\rho}\left[\left(1-\frac{1}{N-M}\right)^{1-\frac{M}{N}}-1\right]}\right\} \tag{60}$$

Besides that, the traffic produced by the unicast is

$$\begin{aligned}R_u^e(D,\Gamma_{us}^e)&=\frac{J\lambda(\rho T_c-D)}{E[k_i]}\cdot(1-\frac{M}{N})\\&=\rho\left(\frac{\rho T_c}{D}-1\right)\cdot(1-\frac{M}{N})\end{aligned} \tag{61}$$

Hence,

$$\begin{aligned}&R^e(D,\Gamma_{us}^e)=R_m^e(D,\Gamma_{us}^e)+R_u^e(D,\Gamma_{us}^e)\\&=\frac{\rho(N-M)}{D}\left(\frac{1-e^{\frac{J\lambda D}{\rho}\left[\left(1-\frac{1}{N-M}\right)^{1-\frac{M}{N}}-1\right]}}{J\lambda M}+\frac{\rho T_c-D}{N}\right)\end{aligned} \tag{62}$$

if $D>JT_c$, we responds each user's request at least every J time slots, which diminishes the inter-transmission-opportunities Under this case, the scheme Γ_{us}^e will degrade to the scheme Γ_u^e and thus shows the same scaling law.

Appendix D
The proof Theorem 5

Consider a same procedure of Theorem 4, we can get

$$R_m^e(D,\Gamma_{cs}^e)=\frac{\rho(N-M)}{J\lambda MD}\left[1-e^{-\frac{J\lambda DM}{\rho N}}\right] \tag{63}$$

Besides that, the traffic produced by the unicast is

$$\begin{aligned}R_u^e(D,\Gamma_{cs}^e)&=\frac{J\lambda(\rho T_c-D)}{E[k_i]}\cdot(1-\frac{M}{N})\\&=\rho\left(\frac{\rho T_c}{D}-1\right)\cdot(1-\frac{M}{N})\end{aligned} \tag{64}$$

Hence,

$$\begin{aligned} R^e(D,\Gamma^e_{cs}) &= R^e_m(D,\Gamma^e_{cs}) + R^e_u(D,\Gamma^e_{cs}) \\ &= \frac{\rho(N-M)}{D}\left(\frac{1-e^{-\frac{J\lambda DM}{\rho N}}}{J\lambda M} + \frac{\rho T_c - D}{N}\right) \end{aligned} \tag{65}$$

if $D > JT_c$, we responds each user's request at least every J time slots, which diminishes the inter-transmission-opportunities Under this case, the scheme Γ^e_{us} will degrade to the scheme Γ^e_u and thus shows the same scaling law.

Appendix E
The Proof of Corollary 1

When $D > JT_c$, it is apparently there exists no gain. Thus we consider another case. Since $\rho T_c - D < T_c$, we can get

$$R^e(D,\Gamma^e_{us}) < \frac{\rho(N-M)}{D}\left(\frac{1}{J\lambda M} + \frac{T_c}{N}\right) \tag{66}$$

For N is large enough and T_c is small enough, we omit the constant term T_c/N, then

$$R^e(D,\Gamma^e_{us}) < \frac{\rho(N-M)}{J\lambda MD} \tag{67}$$

Hence,

$$\frac{R^e(D,\Gamma^e_u)}{R^e(D,\Gamma^e_{us})} > \frac{J}{\rho}, 1 \le \rho \le J \tag{68}$$

For the scheme Γ^e_{cs} and scheme Γ^e_c, the proof is same.

Appendix F
The Proof of Theorem 6

We consider a same procedure in the proof of Theorem 6, then,

$$R^d_{lb}(D) = \sum_{k=0}^{\infty} \frac{(\lambda D)^{k-1} e^{-\lambda D}}{k!} \overline{R_{lb}}(k,P). \tag{69}$$

$\overline{R^d_{lb}}(k,P)$ denotes the lower bound of average traffic under k requests and content popularity distribution P.

Since the size of each content is identical, all request situation is divided into N cases:

$$\begin{aligned} &B_1 : \alpha_i > 0, i = k_1, \alpha_i = 0, i \neq k_1; \\ &B_2 : \alpha_i > 0, i = k_1, k_2, \alpha_i = 0, i \neq k_1, k_2; \\ &\cdots\cdots; \\ &B_N : \alpha_i > 0, i \in \mathbb{K}. \end{aligned}$$

Under the case i, there are only i contents are requested by all users. Based on the results in (18), we can get

$$\overline{R^d_{lb}}(k,P) \ge \frac{cM - c^2 - c^2 M}{M-c} \sum_{i=1}^{N} \mathbb{P}(B_i) \cdot \frac{i}{M} \tag{70}$$

$$= \frac{cM - c^2 - c^2 M}{M(M-c)} \sum_{i=1}^{N} \left(1-(1-p_i)^k\right). \tag{71}$$

Hence,

$$R^d_{lb}(D) = \sum_{k=0}^{\infty} \frac{(\lambda D)^{k-1} e^{-\lambda D}}{k!} \overline{R^d_{lb}}(k,P). \tag{72}$$

$$\ge \frac{cM - c^2 - c^2 M}{M(M-c)} \cdot \frac{1}{\lambda D} \cdot \left(N - \sum_{i=1}^{N} e^{-\lambda D p_i}\right) \tag{73}$$

$$= O\left(\frac{1}{D}\right). \tag{74}$$

Appendix G
The Proof of Theorem 7

We consider a same procedure in the proof of Theorem 1, then,

$$R^d(D,\Gamma^d_u) = \sum_{k=0}^{\infty} \frac{(\lambda D)^{k-1} e^{-\lambda D}}{k!} \overline{R}(k,\Gamma^d_u). \tag{75}$$

Assume that $p_1 \le p_2 \le \cdots p_N$, the average number of users should be transmitted by the shared link is actually $k\left(1-\sum_{i=M+1}^{N} p_i\right)$. We define $N-M$ binary random variable $X_i = 0,1, 1 \le i \le M$, which equals to 1 if and only if the content i is requested by at least one user. Then,

$$\begin{aligned} \overline{R^d}(k,\Gamma^d_u) &= E\left[\sum_{i=1}^{M} X_i\right] = \sum_{i=1}^{M} E\left[X_i\right] \\ &= (N-M)\left(1 - \frac{1}{N-M}\sum_{i=1}^{M}\left(1-p'_i\right)^{1-S_M}\right), \end{aligned} \tag{76}$$

where $p'_i = p_i / \left(1-S_M\right)$ and $S_M = \sum\limits_{i=M+1}^{N} p_i$, then,

$$R^d(D,\Gamma^d_u) = \frac{N-M}{\lambda D}\left\{1 - \sum \frac{e^{\lambda D\left[\left(1-p'_i\right)^{1-S_M}-1\right]}}{N-M}\right\}. \tag{77}$$

Under the condition (8), we can get

$$R^d(D,\Gamma^d_u) = \frac{N-M}{\lambda M}\Theta\left(\frac{1}{D}\right). \tag{78}$$

Appendix H
The Proof of Lemma 3

We take two steps to prove this lemma: we first derive the $R_{\vec{s}}(Q,\Gamma^d_c)$ by a specific request vector that satisfying $\vec{s} = (\alpha_1, \alpha_2, \ldots, \alpha_N)$; we then prove that any request vector satisfying $\vec{s}$ produce the same traffic.

Firstly, consider a request vector $(d_1, d_2, \ldots, d_k)$. Consider a particular bit in one content, termed as content i. Since the prefetching is uniform, this bit has probability $p = C^1_{q_i MF}/C^1_F = q_i M$ of being prefetched in the cache of any fixed user. For any fixed subset of t out of k users, the probability that this bit is prefetched at exactly those t users is

$$(q_i M)^t (1 - q_i M)^{k-t}. \tag{79}$$

Hence, the average number of bits of content i that are cached at exactly those t users is

$$F(q_iM)^t(1-q_iM)^{k-t}. \tag{80}$$

Since $|U/\{k\}| = s-1$, the expected size of $U_{k,S/\{k\}}$ is

$$F(q_iM)^{s-1}(1-q_iM)^{k-s+1} \pm o(F),$$

with high probability. For simplicity, the $o(F)$ term is ignored in the following derivation. Thus, the traffic $R_{\vec{s}}(Q,\Gamma_c^d)$ is

$$\sum_{i=1}^{k} \sum_{v\subset[k],|v|=i} \max_{j\in v}\{(q_{d_j}M)^{i-1}(1-q_{d_j}M)^{k-i+1}\}. \tag{81}$$

Secondly, we prove that any request vector satisfying $\vec{s}$ will produce the same traffic volume. We consider two requested vectors:

$$U_i = (d_1,\dots,d_m,\dots,d_n,\dots,d_k),$$
$$U_i^{'} = (d_1, ,\dots,d_n^* = d_m,\dots,d_m^* = d_n,\dots,d_k).$$

Remark that these two requested vectors satisfying request situation: $\vec{s} = (\alpha_1,\alpha_2,\dots,\alpha_N)$ and the difference between them is that request m and request n exchanges their requested contents. Factually, the different request vectors satisfying the same request situation $\vec{s}$, can be converted to each other via finite exchange. Then, we will show that, under these two requested vectors, the traffic is equal.

The equation (82) refers to the traffic under $U_i = (d_1,\dots,d_m,\dots,d_n,\dots,d_k)$ and the equation (83) refers to the traffic under $U_i' = (d_1,\dots,d_m^*,\dots,d_n^*,\dots,d_k)$.

Consider $d_m^* = d_n$ and $d_n^* = d_m$, we can get

$$\begin{aligned}
&R_m(\vec{s},Q) = R_n^*(\vec{s},Q),\\
&R_n(\vec{s},Q) = R_m^*(\vec{s},Q),\\
&R_{m,n}^*(\vec{s},Q) = R_{n,m}(\vec{s},Q) = R_{m,n}^*(\vec{s},Q),\\
&R_{\o}(\vec{s},Q) = R_{\o}^*(\vec{s},Q).
\end{aligned}$$

Then, $R_{\vec{s}}(Q,\Gamma_c^d) = R_{\vec{s}}^*(Q,\Gamma_c^d)$.

Hence, the request situation can be divided by the $\vec{s}$, and the average traffic is

$$\overline{R^d}(k,Q,\Gamma_c^d) = \sum_{\vec{s}} \mathbb{P}(\vec{s}) \cdot R_{\vec{s}}(Q,\Gamma_c^d). \tag{84}$$

Appendix I
The Proof of Theorem 8

We first prove that $\sum\limits_{i=1}^{N} p_i e^{\lambda DMq_i}$ is a convex function over Q. The Hessian matrix is

$$(\lambda DM)^2 \begin{pmatrix} p_1e^{\lambda DMq_1} & \cdots & 0\\ \vdots & \ddots & 0\\ 0 & \cdots & p_Ne^{\lambda DMq_N} \end{pmatrix} \succ 0.$$

It is strictly positive definite over Q. Then we define a generalized Lagrangian function

$$\begin{aligned}
\mathcal{L}(Q,\sigma,\tau,\kappa) = &\sum_{i=1}^{N} p_i e^{\lambda DMq_i} + \sum_{i=1}^{N}\sigma_i(q_i-\frac{1}{M})\\
&-\sum_{i=1}^{N}\tau_i q_i + \kappa(\sum_{i=1}^{N} q_i - 1).
\end{aligned}$$

Since the objective function is convex, we can get the optimal solution $Q^\dagger$ by the **Karush-Kuhn-Tucker(KKT)** conditions as follows.

$$\begin{aligned}
\frac{\partial}{\partial q_i^\dagger}\mathcal{L}(Q^\dagger,\sigma^*,\tau^*,\kappa^*) &= 0, i=1,\dots,N,\\
\frac{\partial}{\partial \lambda^*}\mathcal{L}(Q^\dagger,\sigma^*,\tau^*,\kappa^*) &= 0,\\
\sigma_i^*(q_i^\dagger - \frac{1}{M}) &= 0, i=1,\dots,N,\\
q_i^\dagger &\le \frac{1}{M}, i=1,\dots,N,\\
\sigma_i^* &\ge 0, i=1,\dots,N,\\
\tau_i^* q_i^\dagger &= 0, i=1,\dots,N,\\
q_i^\dagger &\ge 0, i=1,\dots,N,\\
\tau_i^* &\le 0, i=1,\dots,N,
\end{aligned}$$

$Q^\dagger$ can be derived by the constraints above.

Appendix J
The Proof of Lemma 4

We prove this inequality by the following scaling method.

All request situation is divided into following N cases: $A_i : \alpha_j = 0, j < i, \alpha_j > 0, j \ge i$. In the case A_i, each user only request one of contents $i, i+1,\dots,N$ and their corresponding caching distribution satisfies $q_i \le q_{i+1} \le \cdots \le q_N$. Let $q_i, q_{i+1},\cdots,q_N \leftarrow q_i$. Then, the caching distribution of content $i+1, i+2,\dots,N$ are reduced to q_i. Thus, the traffic rate under this case is

$$\begin{aligned}
&\sum_{s=1}^{K} C_K^s (q_iM)^{s-1}(1-q_iM)^{k-s+1} &(85)\\
&= \frac{1-q_iM}{q_iM}\left(1-(1-q_iM)^k\right). &(86)
\end{aligned}$$

The probability of case A_i is calculated based on multiplication formula. Then,

$$\overline{R^d}(k,Q,\Gamma_c^d) \le \sum_{i=1}^{N}\mathbb{P}(A_i)\frac{1-q_iM}{q_iM}\left[1-(1-q_iM)^k\right]. \tag{87}$$

Thus,

$$\begin{aligned}
&R^d(D,\Gamma_c^d) = \sum_{k=0}^{\infty}\frac{(\lambda D)^{k-1}e^{-\lambda D}}{k!}\overline{R}(k,Q,\Gamma_c^d) &(88)\\
&\le \sum_{k=0}^{\infty}\frac{(\lambda D)^{k-1}e^{-\lambda D}}{k!}\sum_{i=1}^{N}\mathbb{P}(A_i)\frac{1-q_iM}{q_iM}\left[1-(1-q_iM)^k\right] &(89)\\
&= \frac{1}{\lambda D}\sum_{i=1}^{N}\mathbb{P}(A_i)\frac{1-q_iM}{q_iM}\left(1-e^{-\lambda DMq_i}\right). &(90)
\end{aligned}$$

$$R_{\vec{s}}(Q, \Gamma_c^d) = \sum_{i=1}^{K} R_m(\vec{s}, Q) + R_n(\vec{s}, Q) + R_{m,n}(\vec{s}, Q) + R_{\o}(\vec{s}, Q), \tag{82}$$

where

$$\begin{aligned}
R_m(\vec{s}, Q) &= \sum_{v \subset [k], |v|=i, n \in v, m \notin v} \max\{(q_{d_m}M)^{i-1}(1-q_{d_m}M)^{k-i+1}, \max_{j \in v/m}\{(q_{d_j}M)^{i-1}(1-q_{d_j}M)^{k-i+1}\}\},\\
R_n(\vec{s}, Q) &= \sum_{v \subset [k], |v|=i, n \notin v, m \in v} \max\{(q_{d_n}M)^{i-1}(1-q_{d_n}M)^{k-i+1}, \max_{j \in v/n}\{(q_{d_j}M)^{i-1}(1-q_{d_j}M)^{k-i+1}\}\},\\
R_{m,n}(\vec{s}, Q) &= \sum_{v \subset [k], |v|=i, n \notin v, m \in v} \max\left\{ \begin{array}{c} (q_{d_m}M)^{i-1}(1-q_{d_m}M)^{k-i+1}, (q_{d_n}M)^{i-1}(1-q_{d_n}M)^{k-i+1}, \\ \max_{j \in v/n}\{(q_{d_j}M)^{i-1}(1-q_{d_j}M)^{k-i+1}\} \end{array} \right\},\\
R_{\o}(\vec{s}, Q) &= \sum_{v \subset [k], |v|=i} \max_{j \in v/\{m,n\}}\{(q_{d_j}M)^{i-1}(1-q_{d_j}M)^{k-i+1}\}.
\end{aligned}$$

$$R^*_{\vec{s}}(Q, \Gamma_c^d) = \sum_{i=1}^{K} R^*_m(\vec{s}, Q) + R^*_n(\vec{s}, Q) + R^*_{m,n}(\vec{s}, Q) + R^*_{\o}(\vec{s}, Q) \tag{83}$$

where

$$\begin{aligned}
R^*_m(\vec{s}, Q) &= \sum_{v \subset [K], |v|=i, n \in v, m \notin v} \max\{(q_{d_m}M)^{i-1}(1-q_{d_m}M)^{K-i+1}, \max_{j \in v/m}\{(q_{d_j}M)^{i-1}(1-q_{d_j}M)^{K-i+1}\}\},\\
R^*_n(\vec{s}, Q) &= \sum_{v \subset [K], |v|=i, n \notin v, m \in v} \max\{(q_{d_n}M)^{i-1}(1-q_{d_n}M)^{K-i+1}, \max_{j \in v/n}\{(q_{d_j}M)^{i-1}(1-q_{d_j}M)^{K-i+1}\}\},\\
R^*_{m,n}(\vec{s}, Q) &= \sum_{v \subset [K], |v|=i, n \notin v, m \in v} \max\left\{ \begin{array}{c} (q_{d_m}M)^{i-1}(1-q_{d_m}M)^{K-i+1}, (q_{d_n}M)^{i-1}(1-q_{d_n}M)^{K-i+1}, \\ \max_{j \in v/n}\{(q_{d_j}M)^{i-1}(1-q_{d_j}M)^{K-i+1}\} \end{array} \right\},\\
R^*_{\o}(\vec{s}, Q) &= \sum_{v \subset [K], |v|=i} \max_{j \in v/\{m,n\}}\{(q_{d_j}M)^{i-1}(1-q_{d_j}M)^{K-i+1}\}.
\end{aligned}$$

Appendix K
The Proof of Theorem 9

Considering the relaxed solution $Q^\dagger$, D is large enough and condition (8), we can get

$$\begin{aligned}
R^d(D, \Gamma_c^d) &\leq \frac{N-1}{\lambda D} \sum_{i=1}^{N} \mathbb{P}(A_i) \left[1 - \frac{\left(\prod\limits_{j=1}^{N} p_j\right)^{\frac{1}{N}}}{p_i} \right] \qquad (91)\\
&= \frac{N-1}{\lambda D} \left[1 - \left(\prod_{j=1}^{N} p_j\right)^{\frac{1}{N}} \sum_{j=1}^{N} \frac{\mathbb{P}(A_i)}{p_i} \right]. \qquad (92)
\end{aligned}$$

Considering the affect that

$$\sum_{i=1}^{N} \frac{\mathbb{P}(A_i)}{p_i} \geq \frac{\sum\limits_{i=1}^{N} \mathbb{P}(A_i)}{\sum\limits_{i=1}^{N} p_i} \geq 1, \tag{93}$$

Hence,

$$R = \frac{(N-1)\left(1 - \left(\prod\limits_{i=1}^{N} p_i\right)^{\frac{1}{N}}\right)}{\lambda} O\left(\frac{1}{D}\right). \tag{94}$$