Projective Geometry based Coded Caching Schemes with Subexponential and Linear Subpacketizations

by

Hari Hara suthan C, Prasad Krishnan

in

The 19th International Symposium on Communications and Information Technologies (ISCIT 2019)

Ho Chi Minh City, Vietnam

Report No: IIIT/TR/2019/-1

Centre for Communications
International Institute of Information Technology
Hyderabad - 500 032, INDIA
September 2019

Projective Geometry based Coded Caching Schemes with Subexponential and Linear Subpacketizations

Hari Hara Suthan Chittoor, Prasad Krishnan
Signal Processing and Communications Research Center,
International Institute of Information Technology, Hyderabad, India.
Email: {hari.hara@research., prasad.krishnan@}iiit.ac.in

***Abstract*—Coded Caching is a recent technique that optimizes the use of a multi-client broadcast channel by the use of local storage available at the clients and by using coded transmissions to serve multiple clients at once. While large gains in the rate of communication are obtained using coded caching, most existing schemes require that the files at the server be divisible into a large number of parts. In particular, most known coded caching schemes require subpacketization $F = e^{O(K^{\frac{1}{r}})}$, where K is the number of clients and r is some constant positive integer. While few schemes having subpacketization linear in K are known in literature, unfortunately such schemes require large number of users to exist or offer little gain in rate. In this work, we propose a class of coded caching schemes based on projective geometries over finite fields, generalizing recent results. Our construction achieves subexponential (in K) subpacketization, i.e., $F = q^{O((\log_q K)^2)}$, and gain $O((\log_q K)^{n+1})$, for large K and the cached fraction $\frac{M}{N}$ being upper bounded by a constant $\frac{n+1}{q^{\alpha-n}}$(where α, n being positive integer constants such that $n < \alpha$ and q is some prime power). For specific values of the scheme parameters, we get a new linear subpacketization scheme with the number of clients $K \leq q^{2\lambda^2 q^2}$ (and subpacketization F), cache fraction $\frac{M}{N} \leq \lambda$, and coded caching gain $\gamma \geq \frac{4^{\lambda q}}{\lambda q}$ where q is some prime power, and $\lambda \in (0, 1)$.**

***Index Terms*—coded caching, linear subpacketization, broadcast channel, projective geometry.**

I. INTRODUCTION

The present and future wireless communication systems(4G,5G and beyond) are becoming more and more content-centric. The majority of this content is video which is generated well ahead of transmission. Coded caching was proposed in a landmark paper by Ali-Niesen [1] to exploit this aspect of content to reduce the network congestion in peak traffic time by prefetching some of the content in cost effective cache memory available at users during off-peak time. The novel scheme proposed in [1] for broadcast channels were extended to other settings as well (see [2]–[4], for instance).

In [1], the authors considered an error-free broadcast channel with a server containing N files of same size and K users(clients) each having a cache memory capable of storing M files. According to the scheme presented in [1] the system operates in two phases. During *caching phase* (happens in the off-peak time) each file in the server is divided into F equal-sized subfiles (F is known as the *subpacketization* parameter). Each user can store M files (equivalently MF subfiles). During *delivery phase* (happens in the peak time) each user demands a file from the server. Based on the demands and cache contents of users, server makes multiple coded transmissions. The goal is to design the caching and delivery phase so that demands of all users are satisfied. Note that caching phase occurs well ahead of delivery phase, hence demands of users are not available during caching phase. The delivery scheme in [1] serves $\gamma = 1 + \frac{MK}{N}$ users per transmission. The parameter γ is known as the *global caching gain* and the *rate* of the scheme is defined as $R = \frac{K(1-\frac{M}{N})}{\gamma}$. The rate achieved by Ali-Niesen scheme [1] was shown to be optimal in [5] for uncoded cache placement.

Although the scheme [1] is rate-optimal it suffers with the problem of high subpacketization. The subpacketization F of the scheme in [1] is $F = \binom{K}{MK/N}$, which becomes exponential in K as K grows (for constant $\frac{M}{N}$) and hence impractical even for tens of users. The summary of some of the important known schemes is given in Table I. Recently, a class of coded caching schemes achieving subexponential subpacketization were proposed in [6]–[8] (of which the present authors are a subset) using line graphs of bipartite graphs and finite field projective geometry.

Contributions and organization of the paper: In Section II, we formally introduce the system model and review some basic concepts from finite field projective geometry. The main contribution of the present work is presented in Section III and Section V. Section III generalizes the schemes presented in [7], [8]. The advantage we get is more flexibility in choosing system parameters with reduced rate and subpacketization. In Section IV, we analyze the proposed scheme and show that for large K and the cached fraction $\frac{M}{N} \leq \frac{n+1}{q^{\alpha-n}}$ (for some positive integer constants α, n such that $\alpha > n$), we show that our scheme achieves rate $\Theta(\frac{K}{(\log_q K)^{n+1}})$ and subpacketization $q^{O((\log_q K)^2)}$. In Section V, we consider a special class of the scheme proposed in Section III, which achieves linear subpacketization with $F = K$. These schemes allow for a range of parameters, unlike the other known linear subpacketization schemes in literature. In Section VI, we provide numerical examples, which shows that our scheme achieves small subpacketization and reasonable gains.

Notations and Terminology: $\mathbb{Z}^+$ denotes the set of positive integers. We denote the set $\{1, \ldots, n\}$ by $[n]$ for some positive integer n. For sets A, B, the set of elements in A but not in B is denoted by $A \backslash B$. The set of a sized subsets of A is denoted by $\binom{A}{a}$. For $a, b \in \mathbb{Z}^+$ such that $1 \leq a \leq b$, $\binom{a}{b}$ represents the binomial coefficient. The finite field with q elements is $\mathbb{F}_q$. The dimension of a vector space V over $\mathbb{F}_q$ is given as $dim(V)$. For two subspaces V, W, their subspace sum is denoted by $V + W$. Note that $V + W = V \oplus W$ (the direct sum) if $V \cap W = \phi$.

Scheme	$(\mathbf{1}-\frac{\mathbf{M}}{\mathbf{N}})$	Number of Users K	Subpacketization F (for large K and constant $\frac{M}{N}$)	Rate R
Ali-Niesen [1]	$(1-\frac{M}{N})$ for $M<N$ such that $\frac{MK}{N}\in\mathbb{Z}_{\geq 0}$	any K	$c_1e^{Kd_1}$ (for large K)	$\frac{K-t}{1+t}=\frac{K(1-\frac{M}{N})}{\frac{MK}{N}+1}$
Ali-Niesen Scheme with Grouping [9]	Same as [1]	K	$O(c_4e^{d_4K})$ $(c_2,c_3<d_4)$ (for large K)	$\frac{K}{g+1}\left(1-\frac{1}{\lceil\frac{N}{M}\rceil}\right)$, where $g\in\mathbb{Z}$ such that $\frac{K}{g\lceil\frac{N}{M}\rceil}\in\mathbb{Z}$.
Yan et al [10] (PDAs)	$1-\frac{1}{q}$ or $\frac{1}{q}$	Any K	$c_2e^{Kd_2}$ ($d_2<c_2$, for large K)	$\frac{K(1-\frac{M}{N})}{\frac{MK}{N}}$
Shanguan et al [11] (PDAs based on hypergraphs)	$1-\frac{1}{q}$ or $\frac{1}{q}$	Specific choices	For large K $c_3e^{\sqrt{K}d_3}$	$R\approx(2q-1)^2$, such that $q=\frac{\lambda}{2}$, where λ is such that $\frac{M}{N}=\frac{2\lambda-1}{\lambda^2}$
Yan et al [12] (for integers $0<a,b<m$ and $\lambda<min\{a,b\}$ based on strong edge coloring of bipartite graph)	$1-\frac{\binom{a}{\lambda}\binom{m-a}{b-\lambda}}{\binom{m}{a}}$	$\binom{m}{a}$	$\binom{m}{b}$	$\frac{\binom{m}{a+b-2\lambda}\binom{a+b-2\lambda}{a-\lambda}}{\binom{m}{b}}$
Tang et al [13] based on resolvable designs	$1-\frac{1}{q}$ or $\frac{1}{q}$	nq (for some constant q)	For large K, $c_5e^{Kd_5}$, exponent similar to [12] and [11] (some schemes) but less than some schemes of [11]	$\frac{K(1-\frac{M}{N})}{log(F)-c}$ (for some constant c)
Scheme from [14] based on induced matchings of a Rusza Szemeredi graph	$\leq(1-K^{-\epsilon})$ (where $\epsilon=k_1\delta e^{-\frac{k_2}{\delta}}$, for δ as in last column)	K (necessarily large)	K	K^{δ} (some small δ)
PDA scheme P_1 from Cheng et al [15]	For integers k,t $\frac{t+1}{\binom{k}{t}}$	$\binom{k}{t+1}$	$\binom{k}{t}$	$\frac{k}{\binom{k}{t}}$
Two PDA Schemes from [16]	For integers z,q,t $\left(\frac{(q-z)}{q}\right)^t$ and $1-\frac{z}{q}$	$\binom{m}{t}q^t$ and $(m+1)q$	$O\left(q^{\frac{tK^{\frac{1}{t}}}{q}}\right)$	$\left((q-z)/\lfloor\frac{q-1}{q-z}\rfloor\right)^t$ and $(q-z)/\lfloor\frac{q-1}{q-z}\rfloor$

TABLE I: Known results

II. System model and review of finite field projective geometry

We first describe the coded caching system model and then review some basic concepts from projective geometry.

A. System model

Let $K,F,M,N\in\mathbb{Z}^+$. Consider a coded caching system consisting of a server with N equi-popular and equal sized files $\{W_i : i\in[N]\}$ and K users each having a cache memory to store M files $(M\leq N)$. We assume that $K\leq N$, as users are in general much less than the files in the system. Let $\mathcal{K}$ be any set such that $|\mathcal{K}|=K$. We use $\mathcal{K}$ to indicate the set of K users. Let $\mathcal{F}$ be any set such that $|\mathcal{F}|=F$. The coded caching scheme consists of the following two phases.

- **Placement Phase:** During the placement phase each file $W_i, \forall i\in[N]$ is split into F equal sized subfiles denoted as $W_{i,f}$ where $f\in\mathcal{F}$ and $W_{i,f}$ takes values in some Abelian group. Each subfile is of size $1/F$. Each user caches an arbitrary number of subfiles with cache size as its constraint. A caching scheme is *symmetric* if any user caches same number of subfiles of every file.
- **Delivery Phase:** During the delivery phase each user demands a file from the server. Let demand of user $i\in\mathcal{K}$ is W_{d_i}. Based on the user demands and cache contents server broadcasts coded transmissions such that all user demands are satisfied.

The *rate* (R) of such a coded caching scheme is defined as the ratio of the number of bits transmitted to the size of each file, which can be calculated as

$$\text{Rate } R=\frac{\text{Number of transmissions in the delivery phase}}{\text{Number of subfiles in a file}},$$

when each transmission is of the same size as the subfiles. Note that the definition above corresponds to the *peak rate*, i.e., corresponding to the worst case demand scenario in which the K users demand distinct files. The *gain* of a coded caching scheme with rate R is given by the ratio $\frac{K(1-\frac{M}{N})}{R}$, where $K(1-\frac{M}{N})$ denotes the rate of the uncoded transmission scheme.

B. Review of finite field projective geometry [17]

We now review some basic concepts from finite field projective geometry such as counting the number of (fixed dimensional) subspaces of a finite dimensional vector space defined over a finite field. Consider $k,m,q\in\mathbb{Z}^+$ such that q is some prime power and $m\in[k]$. Let $PG_q(k-1,m-1)$ denote the set of all m-dim subspaces of $\mathbb{F}_q^k$, where $\mathbb{F}_q^k$ be a k-dim vector space over a finite field $\mathbb{F}_q$. From [17] (Chapter 3) it is known that $|PG_q(k-1,m-1)|$ is equal to the *q-binomial coefficient* $\begin{bmatrix}k\\m\end{bmatrix}_q$, where $\begin{bmatrix}k\\m\end{bmatrix}_q=\frac{(q^k-1)\dots(q^{k-m+1}-1)}{(q^m-1)\dots(q-1)}$ (where $k\geq m$). Let $\mathbb{T}\triangleq\{T : T\in PG_q(k-1,0)\}$. Let $\theta(k)$ denote the number of distinct 1-dim subspaces of $\mathbb{F}_q^k$. Therefore $\theta(k)=|\mathbb{T}|=\begin{bmatrix}k\\1\end{bmatrix}_q=\frac{q^k-1}{q-1}$. The following lemma and corollary from [8] are used to present our coded caching scheme in Section III-B.

Lemma 1. *[8] Let $k,a,b\in\mathbb{Z}^+$ such that $1\leq a+b\leq k$. Consider a k-dim vector space V over $\mathbb{F}_q$ and a fixed a-dim subspace A of V. The number of distinct (un-ordered) b-sized sets $\{T_1,T_2,\cdots,T_b\}$ such that $T_i\in\mathbb{T},\forall i\in[b]$ and $A\oplus T_1\oplus T_2\oplus\cdots\oplus T_b\in PG_q(k-1,a+b-1)$ is $\frac{\prod_{i=0}^{b-1}(\theta(k)-\theta(a+i))}{b!}$.*

Corollary 1. *[8] Consider two subspaces A, A' of a k-dim vector space V over $\mathbb{F}_q$ such that $A' \subseteq A, dim(A) = a, dim(A') = a-1$. The number of distinct $T \in \mathbb{T}$ such that $A' \oplus T = A$ is $\frac{\theta(a)-\theta(a-1)}{1!} = \frac{q^a - q^{a-1}}{q-1} = q^{a-1}$.*

III. A NEW CODED CACHING SCHEME

In this section we present our projective geometry based coded caching scheme. Before presenting our scheme we first develop some notation which is used to label users, subfiles and transmissions.

A. Mathematical terminology used to describe the scheme:

Consider $k, m, n, t, q \in \mathbb{Z}^+$ such that $m + n + t + 1 \leq k$ and q is some prime power. Consider a k-dim vector space $\mathbb{F}_q^k$. Let L be a fixed $(t-1)$-dim subspace of $\mathbb{F}_q^k$. Consider the following sets of subspaces, where each such subspace contains L.

$$\mathbb{V} \triangleq \{V \in PG_q(k-1, t-1) : L \subseteq V\}.$$
$$\mathbb{R} \triangleq \{R \in PG_q(k-1, n+t-1) : L \subseteq R\}.$$
$$\mathbb{S} \triangleq \{S \in PG_q(k-1, m+t-1) : L \subseteq S\}.$$
$$\mathbb{U} \triangleq \{U \in PG_q(k-1, m+n+t) : L \subseteq U\}.$$

i.e, $\mathbb{V}$ is a set of all t-dim subspaces of $\mathbb{F}_q^k$ containing L. Similarly we have $\mathbb{R}, \mathbb{S}, \mathbb{U}$. Now, consider the following sets, which are used to present our coded caching scheme

$$\mathbb{X} \triangleq \left\{ \{V_1, V_2, \cdots, V_{n+1}\} : \forall V_i \in \mathbb{V}, \sum_{i=1}^{n+1} V_i \in \mathbb{R} \right\}. \quad (1)$$

$$\mathbb{Y} \triangleq \left\{ \{V_1, V_2, \cdots, V_{m+1}\} : \forall V_i \in \mathbb{V}, \sum_{i=1}^{m+1} V_i \in \mathbb{S} \right\}. \quad (2)$$

$$\mathbb{Z} \triangleq \left\{ \{V_1, \cdots, V_{m+n+2}\} : \forall V_i \in \mathbb{V}, \sum_{i=1}^{m+n+2} V_i \in \mathbb{U} \right\}. \quad (3)$$

Thus, $\mathbb{X}$ is the set of all $(n+1)$-sized sets of t-dim subspaces containing L such that there sum is a $(n+t)$-dim subspace. Intuitively each V_i in $\{V_1, V_2, \cdots, V_{n+1}\} \in \mathbb{X}$ has a different extra dimension apart from L. So the dimension of the subspace $\sum_{i=1}^{n+1} V_i$ is $t - 1 + n + 1 = n + t$. Similarly we can understand $\mathbb{Y}, \mathbb{Z}$.

B. Description and verification of our new scheme:

We now describe our coded caching scheme, which includes the caching and delivery schemes, and also present a short verification that the demanded but missing subfiles at the clients are all decodable at respective clients.

Consider a coded caching system with a server consisting of N files. The indexing of users and subfiles are as follows.

Defining the user and subfile indices:

- We index the set of users $\mathcal{K}$ using the set $\mathbb{X}$ as in (1). We use $\mathbb{Y}$ in (2) to denote the subfile indices $\mathcal{F}$.

The placement and delivery phase are as follows.

Placement phase: During the placement phase, we split the file $W_i, \forall i \in [N]$ into subfiles denoted as $W_{i,Y}, \forall Y \in \mathbb{Y}$. The caching is as follows

- For each $X \in \mathbb{X}$, $Y \in \mathbb{Y}$ and $i \in [N]$, the subfile $W_{i,Y}$ is placed in the cache of user X if $X \cup Y \notin \mathbb{Z}$.

Delivery phase: During the delivery phase, each user demands a file from the server. The demand of an arbitrary user $X \in \mathbb{X}$ is denoted by W_{d_X}, where $d_X \in [N]$. Now recall the definition of $\mathbb{Z}$ given in (3). The following describes the transmissions made by the server in the delivery phase.

- For each $Z \in \mathbb{Z}$, the server transmits

$$\bigoplus_{X \in \binom{Z}{n+1}} W_{d_X, Z \setminus X}$$

Observe that every user in $\binom{Z}{n+1}$ demands a subfile which is cached at the remaining users in the set. More precisely consider an arbitrary user $X \in \binom{Z}{n+1}$. Note that $X \in \mathbb{X}$ and $Z \setminus X \in \mathbb{Y}$. Therefore $W_{d_X, Z\setminus X}$ is a valid subfile, and it is demanded by user X since $X \cup (Z \setminus X) \in \mathbb{Z}$. Further any user $X' \subset Z$ caches $W_{d_X, Z\setminus X}$ if $X' \neq X$, by the caching scheme given.

Verification of correct decoding: Consider an arbitrary subfile $W_{d_X, Y}$ which is demanded (and missing) at client X. From the definition of caching scheme, it is easy to see that $X \cup Y \in \mathbb{Z}$. Therefore this subfile is present (uniquely) in the transmission corresponding to $Z = X \cup Y$. Thus every demanded subfile is decoded at the respective client.

The following theorem captures the parameters of our new scheme.

Theorem 1. *For any $k, m, n, t, q, N \in \mathbb{Z}^+$ such that $m + n + t + 1 \leq k$ and q being some prime power, the coded caching scheme in Section III-B has the following parameters.*

$$K = \frac{q^{\frac{n(n+1)}{2}}}{(n+1)!} \prod_{i=0}^{n} \begin{bmatrix} k-t+1-i \\ 1 \end{bmatrix}_q$$

$$F = \frac{q^{\frac{m(m+1)}{2}}}{(m+1)!} \prod_{i=0}^{m} \begin{bmatrix} k-t+1-i \\ 1 \end{bmatrix}_q$$

$$\frac{M}{N} = 1 - q^{(m+1)(n+1)} \prod_{i=0}^{n} \frac{\begin{bmatrix} k-m-t-i \\ 1 \end{bmatrix}_q}{\begin{bmatrix} k-t+1-i \\ 1 \end{bmatrix}_q}$$

$$R = \frac{(m+1)!}{(m+n+2)!} q^{(n+1)(\frac{n}{2}+m+1)} \prod_{i=m+1}^{m+n+1} \begin{bmatrix} k-t+1-i \\ 1 \end{bmatrix}_q$$

$$\gamma = \binom{m+n+2}{n+1}$$

where $K, F, \frac{M}{N}, R, \gamma$ represents number of users, number of subfiles, cache fraction, rate and global caching gain respectively.

Proof:

Recalling $\mathbb{T} \triangleq \{T : T \in PG_q(k-1,0)\}$, we proceed to find our parameters one by one.

Calculating $K = |\mathbb{X}|$: To calculate $|\mathbb{X}|$, we first count the number of distinct sets $\{T_1, T_2, \cdots, T_{n+1}\}$ (such that $T_i \in \mathbb{T},\ \forall i \in [n+1]$ and $L \oplus T_1 \oplus T_2 \oplus \cdots \oplus T_{n+1} \in \mathbb{R}$) which gives distinct $\{L \oplus T_1, L \oplus T_2, \cdots, L \oplus T_{n+1}\} \in \mathbb{X}$. By Lemma 1 we have, the number of distinct sets $\{T_1, T_2, \cdots, T_{n+1}\}$, such that $T_i \in \mathbb{T}$ ($\forall i \in [n+1]$) and $L \oplus T_1 \oplus T_2 \cdots \oplus T_{n+1} \in \mathbb{R}$, is $\frac{\prod_{i=0}^{n}(\theta(k)-\theta(t-1+i))}{(n+1)!}$. It is easy to check that $\{L \oplus T_1, L \oplus T_2, \cdots, L \oplus T_{n+1}\} \in \mathbb{X}$. By Corollary 1 we have, the number of distinct $T \in \mathbb{T}$ such that $L \oplus T = V$ for some fixed $V \in \mathbb{V}$ is q^{t-1}. Therefore for each $\{L \oplus T_1, L \oplus T_2, \cdots, L \oplus T_{n+1}\} \in \mathbb{X}$ there exist $(q^{t-1})^{n+1} = q^{(t-1)(n+1)}$ distinct $\{T'_1, T'_2, \cdots, T'_{n+1}\}$ (where $T'_i \in \mathbb{T}\ \forall i \in [n+1]$) such that $L \oplus T_i = L \oplus T'_i, \forall i \in [n+1]$. Therefore we can write

$$K = \frac{\prod_{i=0}^{n}(\theta(k)-\theta(t-1+i))}{(n+1)!q^{(t-1)(n+1)}} \tag{4}$$
$$= \frac{1}{(n+1)!q^{(t-1)(n+1)}} \prod_{i=0}^{n} \frac{q^k - q^{t-1+i}}{q-1}$$
$$= \frac{q^{(t-1)(n+1)}\left(\prod_{i=0}^{n} q^i\right)}{(n+1)!q^{(t-1)(n+1)}} \prod_{i=0}^{n} \frac{q^{k-t+1-i}-1}{q-1}$$
$$= \frac{q^{\frac{n(n+1)}{2}}}{(n+1)!} \prod_{i=0}^{n} \begin{bmatrix} k-t+1-i \\ 1 \end{bmatrix}_q .$$

Calculating $F = |\mathbb{Y}|$: Proof is similar to that of K (replace n with m).

Calculating $\frac{M}{N}$: Since symmetric caching is considered, each user can cache(equivalently doesn't cache) $\frac{MF}{N}$(equivalently $F - \frac{MF}{N}$) number of subfiles of any file. We obtain $\frac{M}{N}$ by first deriving $F - \frac{MF}{N}$.

Consider an arbitrary user $X \in \mathbb{X}$ and arbitrary file $W_i, i \in [N]$. We know that the subfiles of W_i are $W_{i,Y}, \forall Y \in \mathbb{Y}$. By our caching scheme in Section III-B, to find the number of non-cached subfiles at user X, we have to count the number of $Y \in \mathbb{Y}$ such that $X \cup Y \in \mathbb{Z}$. By following similar proof technique of K, we can write

$$F\left(1 - \frac{M}{N}\right) = \frac{\prod_{i=0}^{m}(\theta(k)-\theta(n+t+i))}{(m+1)!q^{(t-1)(m+1)}}.$$

Since F can be obtained by a technique similar to K by replacing n with m, we have by (4),

$$F = \frac{\prod_{i=0}^{m}(\theta(k)-\theta(t-1+i))}{(m+1)!q^{(t-1)(m+1)}}. \tag{5}$$

Therefore we have,

$$1 - \frac{M}{N} = \prod_{i=0}^{m} \frac{\theta(k)-\theta(n+t+i)}{\theta(k)-\theta(t-1+i)}$$
$$= \frac{\prod_{i=n+1}^{m+n+1}(\theta(k)-\theta(t-1+i))}{\prod_{i=0}^{m}(\theta(k)-\theta(t-1+i))} \tag{6}$$
$$= \frac{\prod_{i=0}^{m+n+1}(\theta(k)-\theta(t-1+i))}{\left(\prod_{i=0}^{m}(\theta(k)-\theta(t-1+i))\right)\left(\prod_{i=0}^{n}(\theta(k)-\theta(t-1+i))\right)}$$
$$= \frac{\prod_{i=m+1}^{m+n+1}(\theta(k)-\theta(t-1+i))}{\prod_{i=0}^{n}(\theta(k)-\theta(t-1+i))} = \prod_{i=0}^{n} \frac{\theta(k)-\theta(m+t+i)}{\theta(k)-\theta(t-1+i)}$$
$$= \prod_{i=0}^{n} \frac{q^k - q^{m+t+i}}{q^k - q^{t-1+i}} = \prod_{i=0}^{n} q^{m+1}\ \frac{q^{k-m-t-i}-1}{q^{k-t+1-i}-1}.$$

Therefore,

$$\frac{M}{N} = 1 - q^{(m+1)(n+1)} \prod_{i=0}^{n} \frac{\begin{bmatrix} k-m-t-i \\ 1 \end{bmatrix}_q}{\begin{bmatrix} k-t+1-i \\ 1 \end{bmatrix}_q}.$$

Calculating R: We know that the number of transmissions in the delivery phase is $|\mathbb{Z}|$. Therefore from the definition of rate we have $R = \frac{|\mathbb{Z}|}{F}$. By using the technique presented in the calculation of K we have

$$|\mathbb{Z}| = \frac{\prod_{i=0}^{m+n+1}(\theta(k)-\theta(t-1+i))}{(m+n+2)!q^{(t-1)(m+n+2)}}.$$

Now by using (5) we get,

$$R = \frac{(m+1)!}{(m+n+2)!} \frac{\prod_{i=m+1}^{m+n+1}(\theta(k)-\theta(t-1+i))}{q^{(t-1)(n+1)}} \tag{7}$$
$$= \frac{(m+1)!}{(m+n+2)!} \frac{1}{q^{(t-1)(n+1)}} \prod_{i=m+1}^{m+n+1} \frac{q^k - q^{t-1+i}}{q-1}$$
$$= \frac{(m+1)!}{(m+n+2)!} q^{(n+1)(\frac{n}{2}+m+1)} \prod_{i=m+1}^{m+n+1} \begin{bmatrix} k-t+1-i \\ 1 \end{bmatrix}_q .$$

Calculating γ: From the definition of global caching gain(γ) we have, $\gamma = \frac{K(1-\frac{M}{N})}{R}$. Now by using $(4), (6), (7)$ we get, $\gamma = \frac{(m+n+2)!}{(m+1)!(n+1)!} = \binom{m+n+2}{n+1}$.

This completes the proof. ∎

Remark 1. *The scheme proposed in Theorem 1 subsumes the schemes proposed in [7], [8] as special cases for* $n = 0, 1$ *respectively.*

IV. Asymptotic analysis of the proposed scheme

In this section, we analyse the asymptotic behaviour of F, R for our coded caching scheme proposed in Section III-B as $\frac{M}{N}$ is upper bounded by a constant and $K \to \infty$. We show that $F = q^{O((\log_q K)^2)}$, while $R = \Theta(\frac{K}{(\log_q K)^{(n+1)}})$. We first recall a bound on q-binomial coefficients that we use.

Lemma 2. *[6] Let* $a, b, \in \mathbb{Z}^+$ *and* q *be some prime power. Then,* $q^{(a-b)b} \leq \begin{bmatrix} a \\ b \end{bmatrix}_q \leq q^{(a-b+1)b}$.

We now continue with our analysis. Throughout our analysis we assume q is a constant and some prime power, and n is some positive integer constant. We now upper bound $\frac{M}{N}$ by a constant. We have by Theorem 1,

$$1-\frac{M}{N}=q^{(m+1)(n+1)}\prod_{i=0}^{n}\frac{\begin{bmatrix}k-m-t-i\\1\end{bmatrix}_q}{\begin{bmatrix}k-t+1-i\\1\end{bmatrix}_q}$$
$$=\prod_{i=0}^{n}\frac{q^{k-t+1-i}-q^{m+1}}{q^{k-t+1-i}-1}\geq\prod_{i=0}^{n}\frac{q^{k-t+1-i}-q^{m+1}}{q^{k-t+1-i}}$$
$$1-\frac{M}{N}\geq\prod_{i=0}^{n}\left(1-\frac{q^i}{q^{k-t-m}}\right).$$

Let $\alpha=k-t-m$. ($\alpha>n$, since $k\geq m+n+t+1$)

$$1-\frac{M}{N}\geq\prod_{i=0}^{n}\left(1-\frac{q^i}{q^\alpha}\right)\geq\prod_{i=0}^{n}\left(1-\frac{q^n}{q^\alpha}\right)$$
$$\geq\left(1-\frac{1}{q^{\alpha-n}}\right)^{n+1}\geq 1-\frac{n+1}{q^{\alpha-n}}. \tag{8}$$

Therefore upper bound on $\frac{M}{N}$ is given as $\frac{M}{N}\leq\frac{n+1}{q^{\alpha-n}}$, where $\alpha=k-t-m$ and n are constants.

We have $K=\frac{q^{\frac{n(n+1)}{2}}}{(n+1)!}\prod_{i=0}^{n}\begin{bmatrix}k-t+1-i\\1\end{bmatrix}_q$. We analyse our scheme as $(k-t)$ grows large (thus K grows large). By Lemma 2 we have,

$$\frac{q^{\frac{n(n+1)}{2}}}{(n+1)!}\prod_{i=0}^{n}q^{k-t-i}\leq K\leq\frac{q^{\frac{n(n+1)}{2}}}{(n+1)!}\prod_{i=0}^{n}q^{k-t+1-i},$$
$$q^{(k-t)(n+1)}\leq(n+1)!K\leq q^{(k-t+1)(n+1)}, \tag{9}$$
$$(k-t)\leq\frac{\log_q((n+1)!K)}{(n+1)}\leq(k-t+1).$$

Hence we have,

$$\frac{\log_q((n+1)!K)}{(n+1)}-1\leq k-t\leq\frac{\log_q((n+1)!K)}{(n+1)}. \tag{10}$$

We now get the asymptotics for the rate. We have, $R=\frac{K(1-\frac{M}{N})}{\gamma}$. From Theorem 1 we have $\gamma=\binom{m+n+2}{n+1}$. Since $\alpha=k-t-m$, we can write $\gamma=\binom{k-t+n-\alpha+2}{n+1}$. We have the following well known bounds on binomial coefficient (e being the base of the natural logarithm) $\left(\frac{a}{b}\right)^b\leq\binom{a}{b}\leq e^b\left(\frac{a}{b}\right)^b$.

By using this result, the bounds on γ can be written as, $\left(\frac{k-t+n-\alpha+2}{n+1}\right)^{n+1}\leq\gamma\leq\left(\frac{e(k-t+n-\alpha+2)}{n+1}\right)^{n+1}$.

By using (10) the lower bound on γ can be written as

$$\left(\frac{\frac{\log_q((n+1)!K)}{(n+1)}+n-\alpha+1}{n+1}\right)^{n+1}\leq\gamma\quad,$$

and the upper bound on γ can be written as

$$\gamma\leq\left(\frac{e\left(\frac{\log_q((n+1)!K)}{(n+1)}+n-\alpha+2\right)}{n+1}\right)^{n+1}.$$

After some simple manipulations we get $\gamma=\Theta\left((\log_q(n+1)!K)^{n+1}\right)=\Theta\left((\log_q K)^{n+1}\right)$. (since n is a constant). Therefore we get $R=\Theta\left(\frac{K}{(\log_q K)^{n+1}}\right)$.

We now obtain the asymptotics for subpacketization F. By K,F expressions in Theorem 1,

$$\frac{F}{K}=\frac{(n+1)!}{(m+1)!}\ \frac{q^{\frac{m(m+1)}{2}}}{q^{\frac{n(n+1)}{2}}}\ \frac{\prod_{i=0}^{m}\begin{bmatrix}k-t+1-i\\1\end{bmatrix}_q}{\prod_{i=0}^{n}\begin{bmatrix}k-t+1-i\\1\end{bmatrix}_q}.$$

By Lemma 2 we have,

$$\frac{F}{K}\leq\frac{(n+1)!}{(m+1)!}\ \frac{\prod_{i=0}^{m}q^i}{\prod_{i=0}^{n}q^i}\ \frac{\prod_{i=0}^{m}q^{k-t+1-i}}{\prod_{i=0}^{n}q^{k-t-i}}$$
$$=\frac{(n+1)!}{(m+1)!}\ \frac{\prod_{i=0}^{m}q^{i-i}}{\prod_{i=0}^{n}q^{i-i}}\ \frac{q^{(k-t+1)(m+1)}}{q^{(k-t)(n+1)}}$$
$$F\leq\frac{(n+1)!\ K}{(m+1)!}\ q^{(k-t+1)m-(k-t)n+1}.$$

By using $m=k-t-\alpha$ we get,

$$F\leq\frac{q^{\log_q((n+1)!K)}\ q^{\left((k-t)^2+(1-\alpha-n)(k-t)+(1-\alpha)\right)}}{(k-t+1-\alpha)!}.$$

By upper bound of (10) we have,

$$\left((k-t)^2+(1-\alpha-n)(k-t)+(1-\alpha)\right)$$
$$\leq\left(\frac{\log_q((n+1)!K)}{(n+1)}\right)^2+(1-\alpha-n)\frac{\log_q((n+1)!K)}{(n+1)}+(1-\alpha).$$

By lower bound of (10) we have,

$$\frac{1}{(k-t+1-\alpha)!}\leq\frac{1}{\lfloor\frac{1}{(n+1)}\log_q((n+1)!K)-\alpha\rfloor!}.$$

By using these bounds we get,

$$F\leq\frac{q^{\frac{1}{(n+1)^2}(\log_q((n+1)!K))^2+\frac{2-\alpha}{n+1}\log_q((n+1)!K)+(1-\alpha)}}{\lfloor\frac{1}{(n+1)}\log_q((n+1)!K)-\alpha\rfloor!}.$$

Using Stirling's approximation for $x!$ as $\sqrt{2\pi x}\left(\frac{x}{e}\right)^x$ for large x, and after some simple manipulations we see that $F=q^{O((\log_q((n+1)!K))^2)}=q^{O((\log_q K)^2)}$ (Since n is a constant).

V. A NEW FLEXIBLE SCHEME WITH LINEAR SUBPACKETIZATION

One of the most interesting regimes for the coded caching problem is that of linear subpacketization, i.e., the case when $F=O(K)$. It is known from [11] via the theory of hypergraphs that linear subpacketization is not sufficient for achieving constant rate. A well known result from [14] shows that there exist coded caching schemes which have $F=K$ and achieve a rate of K^δ (for small δ), and for a small cache fraction. However the number of users required for this construction is extremely high. The Ali-Niesen scheme [1] itself achieves $F = K$ when $\frac{M}{N} = \frac{1}{K}$, but the rate is then $R = \frac{K-1}{2}$. A similar scheme with same parameters is known from [15] (see Section V.B). Thus, most of the existing schemes with linear subpacketization either require extremely large number of users to exist, or have a very small caching gain.

The following corollary to Theorem 1 gives a new linear subpacketization scheme obtained using our projective geometry based technique. Note the scheme is parameterized by the prime power q (finite-field size) and the cache fraction λ, and hence is flexible for different numbers of users and cache fraction.

Corollary 2. *There exists a coded caching scheme with* $F = K \leq q^{2\lambda^2 q^2}$, *cache fraction* $\frac{M}{N} \leq \lambda$, *and coded caching gain* $\gamma \geq \frac{4^{\lambda q}}{\lambda q}$ *(with the rate achieved being* $\frac{K(1-\frac{M}{N})}{\gamma}$*), where* q *is some prime power, and* $\lambda \in (0,1)$ *such that* λq *is a prime power* ≥ 4.

Proof: We choose some specific values for the parameters for our construction in Section III-B to prove this result. From (8), we have that $\frac{M}{N} \leq \frac{n+1}{q}$. We choose $n+1 = \lambda q$ (note that $n+1$ must be an integer and q is a prime power and hence we have our constraints on λ).

From the expressions in Theorem 1, we have that $K = F$ when $m = n$. We choose the least valid value of k, i.e., $k = m+n+t+1$. We thus have with these parameters,

$$K = F \overset{(9)}{\leq} \frac{q^{(k-t+1)(n+1)}}{(n+1)!} = \frac{q^{2(n+1)^2}}{(n+1)!} \leq q^{2\lambda^2 q^2},$$

as stated in the statement of the corollary. We now come to the coded caching gain. From Theorem 1, we have that the global caching gain of the scheme is

$$\gamma = \binom{2n+2}{n+1} \geq \frac{4^{n+1}}{n+1} \geq \frac{4^{\lambda q}}{\lambda q},$$

where the first inequality is a well-known inequality for the middle binomial coefficient that holds for $n \geq 3$. This completes the proof. ■

VI. Numerical Examples

In Table II, we give some numerical examples of the system parameters($K, F, \frac{M}{N}, \gamma = \frac{K(1-\frac{M}{N})}{R}$) obtained by giving valid inputs(k, n, m, t, q) to our scheme presented in Section III-B. The numerical examples show that our scheme is able to achieve low subpacketization(F) with small caching fraction($\frac{M}{N}$) and reasonable global caching gain(γ). For detailed numerical comparisons with other schemes in literature, the readers are referred to prior work by the authors [8], which considers a special case of the general scheme of this paper (with $n = 1$) (Table I in [8]).

$(\mathbf{k},\mathbf{n},\mathbf{m},\mathbf{t},\mathbf{q})$	$\mathbf{K}$	$\mathbf{F}$	$\frac{\mathbf{M}}{\mathbf{N}}$	γ
$(3,0,1,1,3)$	13	78	0.3077	3
$(4,0,1,1,3)$	40	780	0.1000	3
$(3,0,1,1,5)$	31	465	0.1935	3
$(5,1,1,1,2)$	465	465	0.2744	6
$(6,1,1,1,2)$	1953	1953	0.1398	6
$(4,1,0,1,4)$	3570	85	0.0588	3
$(7,1,1,1,2)$	8001	8001	0.0701	6
$(7,2,1,1,2)$	3×10^5	8001	0.1601	10
$(9,2,2,1,2)$	2×10^7	2×10^7	0.0936	20
$(12,3,3,1,2)$	1×10^{13}	1×10^{13}	0.0541	70
$(11,5,0,1,2)$	1×10^{16}	2047	0.0308	7

TABLE II: Some examples for the scheme in Section III-B.

Acknowledgment

This work was supported partly by the Early Career Research Award (ECR/2016/000447) and the MATRICS grant (MTR/2017/000827) from Science and Engineering Research Board (Govt.of India) to Prasad Krishnan. Hari Hara Suthan Chittoor was supported by Visvesvaraya PhD Scheme for Electronics and IT (Govt.of India).

References

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

[2] M. Ji, G. Caire, and A. F. Molisch, "Fundamental limits of caching in wireless d2d networks," *IEEE Transactions on Information Theory*, vol. 62, no. 2, pp. 849–869, Feb 2016.

[3] N. Naderializadeh, M. A. Maddah-Ali, and A. S. Avestimehr, "Fundamental limits of cache-aided interference management," *IEEE Transactions on Information Theory*, vol. 63, no. 5, pp. 3092–3107, May 2017.

[4] A. Sengupta, R. Tandon, and O. Simeone, "Fog-aided wireless networks for content delivery: Fundamental latency tradeoffs," *IEEE Transactions on Information Theory*, vol. 63, no. 10, pp. 6650–6678, Oct 2017.

[5] K. Wan, D. Tuninetti, and P. Piantanida, "On the optimality of uncoded cache placement," in *2016 IEEE Information Theory Workshop (ITW)*, Sep. 2016, pp. 161–165.

[6] P. Krishnan, "Coded caching via line graphs of bipartite graphs," in *2018 IEEE Information Theory Workshop (ITW)*, Nov 2018.

[7] H. H. S. Chittoor, B. M, and P. Krishnan, "Coded caching via projective geometry: A new low subpacketization scheme," in *2019 IEEE International Symposium on Information Theory (ISIT)*, July. 2019, pp. 682–686.

[8] H. H. S. Chittoor and P. Krishnan, "Low subpacketization coded caching via projective geometry for broadcast and d2d networks," *To appear in Proceedings of IEEE Global Communications Conference (GLOBECOM)*, Dec 2019. [Online]. Available: https://arxiv.org/abs/1902.08041.

[9] K. Shanmugam, M. Ji, A. M. Tulino, J. Llorca, and A. G. Dimakis., "Finite-length analysis of caching-aided coded multicasting," *IEEE Transactions on Information Theory*, vol. 62, no. 10, pp. 5524–5537, Oct 2016.

[10] Q. Yan, M. Cheng, X. Tang, and Q. Chen, "On the placement delivery array design for centralized coded caching scheme," *IEEE Transactions on Information Theory*, vol. 63, no. 9, pp. 5821–5833, Sep. 2017.

[11] C. Shangguan, Y. Zhang, and G. Ge, "Centralized coded caching schemes: A hypergraph theoretical approach," *IEEE Transactions on Information Theory*, vol. 64, no. 8, pp. 5755–5766, Aug 2018.

[12] Q. Yan, X. Tang, Q. Chen, and M. Cheng, "Placement delivery array design through strong edge coloring of bipartite graphs," *IEEE Communications Letters*, vol. 22, no. 2, pp. 236–239, Feb 2018.

[13] L. Tang and A. Ramamoorthy, "Coded caching schemes with reduced subpacketization from linear block codes," *IEEE Transactions on Information Theory*, vol. 64, no. 4, pp. 3099–3120, April 2018.

[14] K. Shanmugam, A. M. Tulino, and A. G. Dimakis, "Coded caching with linear subpacketization is possible using ruzsa-szemredi graphs," in *2017 IEEE International Symposium on Information Theory (ISIT)*, June 2017, pp. 1237–1241.

[15] M. Cheng, Q. Yan, X. Tang, and J. Jiang, "Coded caching schemes with low rate and subpacketizations," *arXiv preprint arXiv:1703.01548*, 2017.

[16] M. Cheng, J. Jiang, Q. Yan, and X. Tang, "Constructions of coded caching schemes with flexible memory size," *IEEE Transactions on Communications*, vol. 67, no. 6, pp. 4166–4176, June 2019.

[17] J. Hirschfeld, *Projective Geometries Over Finite Fields. Oxford Mathematical Monographs.* Oxford University Press New York, 1998.