Coded Caching via Line Graphs of Bipartite Graphs

by

prasad.krishnan

in

Information and Coding Theory 2018 (ITW2018)

Report No: IIIT/TR/2018/-1

Centre for Communications
International Institute of Information Technology
Hyderabad - 500 032, INDIA
November 2018

Coded Caching via Line Graphs of Bipartite Graphs

Prasad Krishnan
International Institute of Information Technology, Hyderabad,
Email: prasad.krishnan@iiit.ac.in

Abstract—**We present a coded caching framework using line graphs of bipartite graphs. A clique cover of the line graph describes the uncached subfiles at users. A clique cover of the complement of the square of the line graph gives a transmission scheme that satisfies user demands. We then define a specific class of such caching line graphs, for which the subpacketization, rate, and uncached fraction of the coded caching problem can be captured via its graph theoretic parameters. We present a construction of such caching line graphs using projective geometry. The presented scheme has a rate bounded from above by a constant with subpacketization level $q^{O((log_q K)^2)}$ and uncached fraction $\Theta(\frac{1}{\sqrt{K}})$, where K is the number of users and q is a prime power. We also present a subpacketization-dependent lower bound on the rate of coded caching schemes for a given broadcast setup.**

Due to space considerations, we provide the full version of the paper containing the proofs of the claims, more examples, better literature survey in [1].

I. Introduction

Caching has been in vogue to lay off the traffic during peak times in a communication network by storing part of the information demanded by users (clients) in local storage known as *caches*. In this way, during the peak hours, the server can transmit only the non-cached information thus reducing the traffic. For instance, consider the setting taken in [2], consisting of a single-server error-free broadcast channel with K clients(users), N files at the server (each file comprised of F subfiles, where F is known as the *subpacketization* parameter), with each client capable of caching MF subfiles. By populating the cache during the caching phase when the demand requests are not present, traditional caching can achieve a rate R equal to $K(1-\frac{M}{N})$ during the delivery phase when the network is required to satisfy the demands (the rate R is defined as fraction of the number of transmitted packets (each of size equal to a subfile) to F).

The paradigm of Coded Caching, introduced in [2], was based on the idea of transmitting coded packets during the peak times to further reduce the network usage. In contrast to the uncoded caching scenario which required a rate $K(1-\frac{M}{N})$, the coded caching scheme shown in [2] achieves a rate $\frac{K(1-\frac{M}{N})}{1+\frac{MK}{N}}$, which is constant as $K\to\infty$ for constant $\frac{M}{N}$. This is achieved by designing both the caching phase and the delivery phase carefully. The tremendous rate advantage shown in [2] were shown in other settings also, for instance [3], [4]. The scheme in [2] is also shown to be order optimal, i.e., within a constant multiple of the optimal rate for the same set of parameters. The scheme of [2] is achieved by dividing each file into $F=\binom{K}{\frac{MK}{N}}$ subfiles, and caching them appropriately in the client caches. It was noticed in [5] that for this scheme, the subpacketization required is exponential for constant $\frac{M}{N}$ as K grows large (as $\binom{K}{Kp}\approx 2^{KH(p)}$ for constant $0<p<1$, where $H(p)$ is the binary entropy). Since then a number of papers [5]–[12] have presented new schemes for coded caching which use smaller subpacketization at the cost of having increased rate or cache requirement compared to [2]. Many of the schemes presented in these prior works require exponential subpacketization (in K, for large K) to achieve a constant rate. The subpacketization of particular schemes of [8], [12] have been shown to be sub-exponential, while some schemes of [11] have subpacketization that is linear or polynomial (in K) at the cost of either requiring larger cache M or larger rate compared to [2]. Interestingly, a linear subpacketization scheme ($F=K$) was shown in [9] using a graph theoretic construction with near constant rate and small memory requirement. However the construction in [9] holds for very large values of K only. In [8], it was shown that subpacketization linear in K is impossible if we require constant rate.

The contributions and organization of this work are as follows. After reviewing the work of [6] in Section II, we prove a lower bound on the peak delivery rates of coded caching schemes using the properties of an associated bipartite graph (Section III). We then map the problem of finding a valid transmission scheme corresponding to a bipartite caching scheme to a clique cover problem of a graph derived from the line graph of the bipartite graph (Section IV). In Section IV, we also show that the existence of a class of such line graphs of bipartite graphs implies the existence of coded caching schemes for which there is a nice characterization of the rate, uncached fraction, and subpacketization. We then give a coded caching scheme using a construction of such caching line graphs based on projective geometries over finite fields (Section V). The first row of Table I lists the actual parameters of our scheme (the entries are expressed as q-binomial coefficients). The other two rows indicate asymptotic results as $K\to\infty$ (with constant field size q) respectively. We note that the last row shows a constant rate with sub-exponential packetization achieved by the scheme in this paper. We conclude the paper with a short discussion in Section VI.

Notations and Terminology: For a positive integer n, we denote by $[n]$ the set $\{1,\ldots,n\}$. We recall only minimal facts regarding graph theory. For other standard definitions, the reader is referred to [13]. A graph G consists of a set $V(G)$ of vertices and a set $E(G)\subset\{\{u,v\}:u,v\in V(G)\}$ of edges. For a subset A of vertices of graph G, we denote $\mathcal{N}(A)$ as the set of adjacent vertices of A. A *bipartite graph* B is one whose edges can be visualized as being between

978-1-5386-3599-5/18/$31.00©2018 IEEE

$1-\frac{M}{N}$	K	F	R
For non-negative integers k,m,t with $1 \leq m+t \leq k$			
$\frac{\begin{bmatrix} m+t \\ t \end{bmatrix}_q}{\begin{bmatrix} k \\ t \end{bmatrix}_q}$	$\begin{bmatrix} k \\ t \end{bmatrix}_q$	$\begin{bmatrix} k \\ m+t \end{bmatrix}_q$	$\frac{\begin{bmatrix} m+t \\ t \end{bmatrix}_q}{\begin{bmatrix} k-m \\ t \end{bmatrix}_q}$
Limiting behaviors as k grows, for constants $t, k-m, q$:			
$\geq q^{(m+t-k-1)t}$ (constant)	$\begin{bmatrix} k \\ t \end{bmatrix}_q$	$\leq K^{\frac{k-t-m+1}{t}}$ ($O(poly(K))$)	$\leq \frac{K}{q^{2(k-m-t-1)t}}$ ($\Theta(K)$)
Limiting behaviors as k grows, for constants $t, k-2m, q$:			
$\Theta(\frac{1}{\sqrt{K}})$	$\begin{bmatrix} k \\ t \end{bmatrix}_q$	$q^{O((log_q K)^2)}$	$\leq q^{(2m+t-k+1)t}$ (constant)

TABLE I: Parameters of Scheme in Section V. (proofs of last two rows are available in [1])

two subsets of a partition of the vertex set (called *left* and *right* vertices of B). A subset $S \subseteq V$ is called a *clique* of G if all vertices in S are adjacent to each other (we assume vertices to be cliques of size 1). A *b-clique-cover* of G is a collection of cliques $D_i : i = 1, \ldots, b$ such that $\dot\cup_i D_i = V$, where $\dot\cup$ indicates a disjoint union.

II. Bipartite Graph based Coded Caching and Delivery based on [6]

Let $\mathcal{K}$ be the set of users(clients) ($|\mathcal{K}| = K$) in a system consisting of one server having files $\{W_i : i \in [N]\}$ connected to the clients via a error-free broadcast channel. Let F be the subpacketization level, i.e. each file is composed of F subfiles, each taking values according to a uniform distribution from some finite abelian group $\mathcal{A}$. The subfiles of file W_i are denoted as $W_{i,f} : f \in \mathcal{F}$ for some set $\mathcal{F}$ of size F. Let MF denote the number of subfiles that can be stored in the cache of any user. A *coded caching scheme* consists of two subschemes (as in [2]), a *caching scheme* according to which subfiles of the files are placed in the user caches during periods when the traffic is low, and a *transmission scheme* that consists of broadcast transmissions from the server satisfying the demands of the clients appearing during the demand phase. We assume *symmetric caching* throughout the paper, i.e., the caches at the users are populated in such a way that if user $k \in \mathcal{K}$ caches the subfile $f \in \mathcal{F}$ of any file, then it caches the subfile $f \in \mathcal{F}$ of each file. All the schemes presented in [5]–[12] employ symmetric caching. We also assume throughout this work that $\frac{MF}{N}$ is an integer which is the number of subfiles of any particular file stored in a user's cache. In the delivery scheme, the transmissions (each of size equal to that of any subfile) must be done so that the demands of the clients are all satisfied. As in [2], the rate R of the coded caching scheme is defined as

$$\text{Rate } R = \frac{\text{Number of transmissions in the transmission scheme}}{\text{Number of subfiles in a file}}.$$

We can visualize the symmetric caching scheme (with fully populated caches) using a bipartite graph, following [6]. Consider a bipartite graph B with $\mathcal{K}$ being the left(user) vertices and the right(subfile) vertices being $\mathcal{F}$. We then define the edges of the bipartite graph to denote the uncached subfiles of the files, i.e, for $k \in \mathcal{K}, f \in \mathcal{F}$, an edge $\{k, f\} \in E(B)$ exists if and only if user k does *not* contain in its cache the subfile f of each file. Clearly, this bipartite graph is left-regular, with $F\left(1 - \frac{M}{N}\right)$ being the degree of any user vertex. Indeed any left-regular bipartite graph defines a caching scheme, which we formalize below.

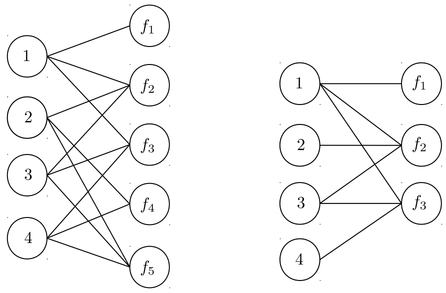

Fig. 1: The left figure is a bipartite caching scheme with 4 users and 5 subfiles with $1 - M/N = 3/5$. Edges indicate missed subfiles. The right figure shows the subgraph induced by $\mathcal{N}(1) \cup \mathcal{N}(\mathcal{N}(1))$.

Definition 1 (Bipartite Caching Scheme)**.** *Given a bipartite D-left-regular graph with K left vertices and F right vertices denoted by $B(K, D, F)$ (or in short, B), the symmetric caching scheme defined on K users with subpacketization F with the edges of B indicating the uncached subfiles at the users, is called the (K, D, F) bipartite caching scheme associated with the bipartite graph B.*

Remark 1. *We observe that the bipartite caching scheme associated with the graph $B(K, D, F)$ has the uncached fraction $1 - \frac{M}{N} = \frac{D}{F}$.*

Fig. 1 shows a graph describing a $(4, 3, 5)$ bipartite caching scheme. Note that during the caching phase the user demands are not available. We now look at the transmission phase during which the user $k \in \mathcal{K}$ demands one file W_{d_k}(for some $d_k \in [N]$), as given in [6]. An *induced matching* $\mathcal{M}$ of a graph G is a matching such that the induced subgraph of the vertices of $\mathcal{M}$ is $\mathcal{M}$ itself. For an induced matching $\mathcal{M}$ of B consisting of edges $\{\{k_j, f_j\} : j \in [l]\}$, consider the associated transmission $Y_{\mathcal{M}} = \sum_{j=1}^{l} W_{d_{k_j}, f_j}$. As $\mathcal{M}$ is an induced matching, $W_{d_{k_j}, f_j}$ is a subfile unavailable but demanded at user k_j. By the same reason, each user k_j has all the subfiles involved in $Y_{\mathcal{M}}$ in its cache except for $W_{d_{k_j}, f_j}$, hence user k_j can decode $W_{d_{k_j}, f_j}, \forall j \in [l]$. A *b-strong-edge-coloring* of a graph is an assignment of labels (called *colors*) from a finite set $\mathcal{C}$ of size b to each of its edges such that the set of all edges of any color (called a *color class*) form an induced matching. Let $\{M_j, j \in [n]\}$ be the set of all induced matchings (color classes) arising from a strong edge coloring of B. It is not difficult to see that the transmissions $Y_{M_j} : j \in [n]$ (constructed as before) corresponding to $M_j : j \in [n]$ satisfies the demands of all the users. The rate of this transmission scheme is then $\frac{n}{F}$.

III. Lower bound on rate of delivery scheme for symmetric caching

In this section, we show a bound on the rate of the transmission scheme associated with a $(K, D = F(1 - \frac{M}{N}), F)$

bipartite caching scheme associated with B. As $W_{i,f}$ takes values from $\mathcal{A}$ with uniform distribution, taking the base of logarithm as $|\mathcal{A}|$, we have the Shannon entropy of $W_{i,f}$ as $H(W_{i,f}) = 1, \forall i, f$. Thus $H(W_i) = F, \forall i$. For a given (K, D, F) bipartite caching scheme, a rate R is said to be *achievable* if there exists some transmission scheme with rate R that satisfies all client demands. We now prove a lower bound on the infimum R^* of all achievable rates for a given (K, D, F) bipartite caching scheme.

Theorem 1. *Let k be any left-vertex (user vertex) of B and let H be the subgraph of B induced by the vertices $\mathcal{N}(k) \cup \mathcal{N}(\mathcal{N}(k))$. Let $N' = min(N, |\mathcal{N}(\mathcal{N}(k))|)$. Let $U = \{k_j : j \in [N']\}$ be a subset of N' vertices of $\mathcal{N}(\mathcal{N}(k))$ taken in some order such that $k_1 = k$. For $j \in [N']$, let ρ_j be the set of right vertices (subfiles) in H which are adjacent to $\{k_i : i \in [j]\}$. Let R^* be the infimum of all achievable rates for the bipartite caching scheme defined by B. Then $R^*F \geq \sum_{j=1}^{N'} \rho_j$.*

In particular, we must have

$$R^*F \geq min\left((K+F)\left(1-\frac{M}{N}\right), F\left(1-\frac{M}{N}\right)+N\right)-1. \tag{1}$$

Proof: Available in [1]. ■

Example 1. *The figure on the right in Fig. 1 shows the subgraph induced by $\mathcal{N}(1) \cup \mathcal{N}(\mathcal{N}(1))$ of the bipartite caching graph $B(4,3,5)$ on the left. Assuming the number of files $N \geq 4$, following Theorem 1, we can take $\rho_1 = 3, \rho_2 = 2, \rho_3 = 1, \rho_4 = 0$ (where the user vertices are taken in the order $1, 3, 2, 4$). We then get the minimum rate $R^* \geq \frac{\sum_{i=1}^{4} \rho_i}{5} = \frac{6}{5}$.*

IV. Line Graphs of Bipartite Graphs and Caching

In this section, we shall map the coded caching problem to the line graph of the bipartite caching graph B described in the previous section. The line graph $\mathcal{L}(G)$ of a graph G is a graph in which the vertex set $V(\mathcal{L}(G))$ is the edge set $E(G)$ of G, and two vertices of $V(\mathcal{L}(G))$ are adjacent if and only if they share a common vertex in G. The square of a graph G is a graph G^2 having $V(G^2) = V(G)$, and an edge $\{u, v\} \in E(G^2)$ if and only if either $\{u, v\} \in E(G)$ or there exists some $v_1 \in V(G)$ such that $\{u, v_1\}, \{v_1, v\} \in E(G)$. The following result is folklore and easy to prove.

Lemma 1. *There exists a b-clique-cover for $\overline{\mathcal{L}^2}$ if and only if there exists a b-strong-edge-coloring for G , with the cliques in the clique cover of $\overline{\mathcal{L}^2}$ corresponding to the color classes (induced matchings) arising from the strong edge coloring of G.*

By Lemma 1 and Section II, a valid transmission scheme corresponding to the caching scheme associated with B can be obtained by obtaining a clique cover for $\overline{\mathcal{L}^2(B)}$. From the arguments in Section II, such a transmission scheme will involve one transmission per each clique in a clique cover of $\overline{\mathcal{L}^2}$. Fig. 2 shows the graph $\overline{\mathcal{L}^2}$ for the line graph of the bipartite graph shown in Fig. 1. A clique cover consisting of 6 cliques is also shown, each containing 2 vertices. Thus the number of transmissions is 6, and the rate is $\frac{6}{5}$, which is optimal following the lower bound in Example 1.

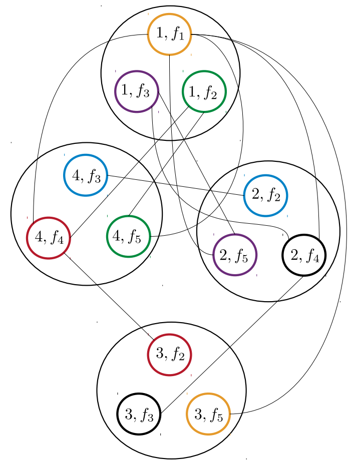

Fig. 2: The graph $\overline{\mathcal{L}^2}$ corresponding to the bipartite graph in Fig. 1. The same-coloured vertices correspond to cliques.

It turns out that the line graph of the left-regular bipartite graph B is highly structured, and any such structured graph will serve as a line graph of such a bipartite graph.

Proposition 1. *A graph $\mathcal{L}$ containing KD vertices is the line graph of a D-left-regular bipartite graph $B(K, D, F)$ if and only if the following conditions are satisfied.*

(C1) *The vertices of $\mathcal{L}$ can be partitioned into K disjoint cliques containing D vertices each. We denote these cliques by $\mathcal{U}_k : k \in [K]$ and call them as the user-cliques. We label the vertices of $\mathcal{U}_k$ as $e_{k,i} : i \in [D]$.*

(C2) *Consider distinct $k_1, k_2 \in [K]$. For any $e_{k_1,i} \in \mathcal{U}_{k_1}$, there exists at most one vertex $e_{k_2,j} \in \mathcal{U}_{k_2}$ such that $\{e_{k_1,i}, e_{k_2,j}\} \in E(\mathcal{L})$.*

(C3) *For any k and any vertex $e_{k,i} \in \mathcal{U}_k$, the set $\{e_{k,i} \cup \mathcal{N}(e_{k,i}) \backslash \mathcal{U}_k\}$ containing $e_{k,i}$ and all adjacent vertices of $e_{k,i}$ except those in $\mathcal{U}_k$, forms a clique. We refer to these cliques as the subfile-cliques. Let r be the number of subfile-cliques in $\mathcal{L}$ and the subfile-cliques be denoted as $\mathcal{S}_i : i \in [r]$.*

(C4) *The number of right vertices of B is*

$$F = KD - \sum_{i=1}^{r}(|\mathcal{S}_i| - 1) = r. \tag{2}$$

Proof: Available in [1]. ■

By Proposition 1, if we construct a graph $\mathcal{L}$ satisfying conditions (C1)-(C3), then we have constructed a caching scheme based on a bipartite graph B such that $\mathcal{L}(B) = \mathcal{L}$ with subpacketization F as in (2). We therefore give the following definition.

Definition 2. *A graph $\mathcal{L}$ is called a caching line graph if it satisfies conditions (C1)-(C3) of Proposition 1 for some parameters K and D.*

Henceforth all our line graphs are caching line graphs. By Lemma 1, any clique cover of $\overline{\mathcal{L}^2}$ (the complement of the square of $\mathcal{L}$) gives us a transmission scheme (one transmission per clique) that satisfies all receiver demands.

In order to obtain a clique cover of $\overline{\mathcal{L}^2}$, we have to understand the behaviour of the cliques of $\overline{\mathcal{L}^2}$.

Lemma 2. *A subset of vertices $\mathcal{C} \subset V(\overline{\mathcal{L}^2})$ is a clique of $\overline{\mathcal{L}^2}$ if and only if the following condition is true.*

- *For any two vertices $e_{k_1,i_1}, e_{k_2,i_2} \in \mathcal{C}$, there exists no vertex in $\mathcal{U}_{k_2}$ adjacent to e_{k_1,i_1} in $\mathcal{L}$.*

Furthermore, any clique of $\overline{\mathcal{L}^2}$ contains at most one vertex from each of the user-cliques of $\mathcal{L}$.

Proof: Available in [1]. ■

We now define a specific class of caching line graphs called (c,d)-caching line graphs. The reason for considering (c,d)-caching line graphs is because they yield easily to the computation of the rate and the subpacketization, as Theorem 2 will show.

Definition 3. *A caching line graph $\mathcal{L}$ such that $\mathcal{L}$ has a clique cover consisting of c-sized disjoint subfile cliques and $\overline{\mathcal{L}^2}$ has a clique cover consisting of d-sized disjoint cliques, for some positive integers c, d, is called a (c,d)-caching line graph.*

Theorem 2. *Consider a (c,d)-caching line graph $\mathcal{L}$. Then there is a coded caching scheme consisting of the caching scheme given by $\mathcal{L}$ with $F = \frac{KD}{c}$ (and thus $\frac{M}{N} = 1-\frac{c}{K}$), and the associated transmission scheme based on a clique cover of $\overline{\mathcal{L}^2}$ having rate $R = \frac{c}{d}$. Furthermore, if the number of files $N \geq K$, the rate R of this scheme satisfies $R \leq \frac{R^*+\frac{1}{F}}{d(\frac{1}{F}+\frac{1}{K})}$, where R^* is the infimum of all achievable rates for L with subpacketization $F = \frac{KD}{c}$.*

Proof: Available in [1]. ■

In [1], we reinterpret some priorly known coded caching schemes from [2], [7], [10] as schemes based on caching line graphs. In the forthcoming section, we present a new explicit construction of a caching scheme based on (c,d)-caching line graphs.

V. A Line Graph based Coded Caching Scheme based on Projective Geometry

We recollect some basic ideas of projective geometries over finite fields. The reader is referred to [14] for more details. For positive integers k, let $PG_q(k-1)$ denote the $(k-1)$-dimensional projective space over $\mathbb{F}_q$. The elements of $PG(k-1,q)$ are called the *points* of $PG_q(k-1)$. The points of $PG_q(k-1)$ can be considered as the representative vectors of one-dimensional subspaces of $\mathbb{F}_q^k$. For $m \geq 1, 1 \leq m \leq k$, let $PG_q(k-1, m-1)$ denote the set of m-dimensional subspaces of $\mathbb{F}_q^k$. It is known that $|PG_q(k-1,m-1)|$ is equal to the *Gaussian binomial coefficient* or the *q-binomial coefficient*, $\begin{bmatrix} k \\ m \end{bmatrix}_q$, where $\begin{bmatrix} k \\ m \end{bmatrix}_q = \frac{(q^k-1)\ldots(q^{k-m+1}-1)}{(q^m-1)\ldots(q-1)}$. In any Gaussian binomial coefficient $\begin{bmatrix} a \\ b \end{bmatrix}_q$ given in this paper we assume that $1 \leq b \leq a$. The following is known about the Gaussian binomial coefficients (for example, see [14] for a proof).

Lemma 3. *The Gaussian binomial coefficient $\begin{bmatrix} k \\ m \end{bmatrix}_q$ is the number of subspaces of dimension m of any k-dimensional subspace over $\mathbb{F}_q$. Also, $\begin{bmatrix} k \\ m \end{bmatrix}_q = \begin{bmatrix} k \\ k-m \end{bmatrix}_q$. The number of elements of $PG_q(k-1,m-1)$ that contain a given t-dimensional subspace ($1 \leq t \leq m$) is*

$$\frac{(q^{k-t}-1)\ldots(q^{k-m+1}-1)}{(q^{m-t}-1)\ldots(q-1)} = \begin{bmatrix} k-t \\ m-t \end{bmatrix}_q.$$

In the following subsection, we give a construction of a coded caching scheme based on projective geometry. The scheme we present can be thought of a q-analogue of a generalization of the original scheme of [2]. We first give the caching line graph $\mathcal{L}$ by describing its user-cliques and subfile-cliques (each of same size), and then show that there is a clique cover of $\overline{\mathcal{L}^2}$ containing cliques of the same size.

A. Coded Caching Scheme Construction

Consider positive integers k, m, t such that $m + t \leq k$. Let $K = \begin{bmatrix} k \\ t \end{bmatrix}_q$. We first initialize $\mathcal{L}$ by its user-cliques. The user-cliques are identified by t-dimensional subspaces of $\mathbb{F}_q^k$. For each t-dimensional subspace V of $\mathbb{F}_q^k$, create the vertices corresponding to the user-clique identified by V,

$$\mathcal{C}_V = \{(V,X), \forall X \in PG_q(k-1, m+t-1) : V \subseteq X\}.$$

Thus, $D = |\mathcal{C}_V| = \begin{bmatrix} k-t \\ (m+t)-t \end{bmatrix}_q = \begin{bmatrix} k-t \\ m \end{bmatrix}_q$ by Lemma 3. For each $(m+t)$-dimensional subspace X of $\mathbb{F}_q^k$, we construct the subfile clique of $\mathcal{L}$ associated with X as

$$\mathcal{C}_X = \{(V,X) \in V(\mathcal{L}) : \forall V \text{ such that } V \subseteq X\}.$$

It's not difficult to see that the cliques $\{\mathcal{C}_X : X \in PG_q(k-1, m+t-1)\}$ partition $V(\mathcal{L})$.

In order to decide on the transmission scheme, we have to obtain a clique cover of $\overline{\mathcal{L}^2}$. The clique cover of $\overline{\mathcal{L}^2}$ that we wish to obtain is based on a relabelling of the vertices of $\mathcal{L}$ based on m-dimensional subspaces of $\mathbb{F}_q^k$. Towards that end, we first require the following lemmas(Lemma 4 and Lemma 5) using which we can find 'matching' labels to the t-dimensional and m-dimensional subspaces of some $X \in PG_q(k-1, m+t-1)$.

Lemma 4. *Consider some element $X \in PG_q(k-1, m+t-1)$. Let $\left\{ V_i, i = 1, \ldots, \begin{bmatrix} m+t \\ t \end{bmatrix}_q \right\}$ denote the t-dimensional subspaces of X taken in some fixed order. Then the set of m-dimensional subspaces of X can be written as an ordered set as $\left\{ T_i,\ i = 1, \ldots, \begin{bmatrix} m+t \\ m \end{bmatrix}_q \right\}$ such that $T_i \oplus V_i = X, \forall i$ (where $\oplus$ denotes direct sum). Moreover such an ordering can be found in operations polynomial in $\begin{bmatrix} m+t \\ t \end{bmatrix}_q$.*

Proof: See [1]. ■

For a t-dimensional space V_i contained in a $(m+t)$-dimensional space X, let T_i (the m-dimensional subspace as obtained in Lemma 4 such that $T_i \oplus V_i = X$) be called *the matching subspace of V_i in X*. Using these matching subspaces, we can obtain an alternate labeling scheme for the vertices of our caching line graph $\mathcal{L}$. The alternate labels are given as follows.

- Let the alternate label for (V, X) be $(V, T_{V,X})$, where $T_{V,X}$ is the m-dimensional matching subspace of V in X obtained using Lemma 4.

The following lemma ensures that the alternative labeling given above is indeed a valid labelling, i.e., it uniquely identifies the vertices of $\mathcal{L}$.

Lemma 5. *No two vertices of $V(\mathcal{L})$ have the same alternate label, i.e., if $(V_1, X_1), (V_2, X_2) \in V(\mathcal{L})$ have the same alternate label $(V, T_{V,X})$, then $(V_1, X_1) = (V_2, X_2)$.*

Proof: See [1]. ∎

We are now in a position to present the clique-cover of $\overline{\mathcal{L}^2}$. Our cliques are represented in terms of the alternate labels given to the vertices of $\mathcal{L}$. We first show the structure of one such clique.

Lemma 6. *For a m-dimensional subspace T of $\mathbb{F}_q^k$, consider the set of vertices of $\overline{\mathcal{L}^2}$ (identified by their alternate labels) as follows.*

$$\mathcal{C}_T = \{(V, T) \in V(\mathcal{L}) : V \in PG_q(k-1, t-1)\}.$$

Then $\mathcal{C}_T$ is a $\begin{bmatrix} k-m \\ t \end{bmatrix}_q$-sized clique of $\overline{\mathcal{L}^2}$.

Proof: See [1]. ∎

In [1] (Lemma 7), we show that the cliques $\{\mathcal{C}_T : T \in PG_q(k-1, m-1)\}$ partition $V(\mathcal{L})$. We hence have the following theorem which summarizes our construction. The proof of the theorem follows from Theorem 2.

Theorem 3. *The caching line graph $\mathcal{L}$ constructed in Section V-A is a $\left(\begin{bmatrix} m+t \\ t \end{bmatrix}_q, \begin{bmatrix} k-m \\ t \end{bmatrix}_q\right)$-caching line graph and defines a coded caching scheme with $K = \begin{bmatrix} k \\ t \end{bmatrix}_q$, $F = \begin{bmatrix} k \\ m+t \end{bmatrix}_q$, $\frac{M}{N} = 1 - \frac{\begin{bmatrix} m+t \\ t \end{bmatrix}_q}{\begin{bmatrix} k \\ t \end{bmatrix}_q}$, and $R = \frac{\begin{bmatrix} m+t \\ t \end{bmatrix}_q}{\begin{bmatrix} k-m \\ t \end{bmatrix}_q}$.*

Proof: See [1]. ∎

In [1] (Section VI), we analyze our presented scheme and show that it achieves the parameters as shown in Table I. Table II gives a comparison of the actual quantities between our scheme and that of [2] for some particular choice of users and the uncached fraction (as defined by our scheme).

VI. Conclusion

In this work, we have presented a framework for constructing coded caching schemes for broadcast networks via line graphs of bipartite graphs, building on results from [6]. We then present one explicit construction of such a (c, d)-caching line graph using projective geometry. For the uncached fraction $(1 - \frac{M}{N})$ lower bounded by a constant, this scheme achieves subpacketization $F = O(poly(K))$, however the rate R is $O(K)$. In another regime of operation where the rate remains below a constant, we get $F = q^{O(log_q K)^2}$ while the uncached fraction $1 - \frac{M}{N}$ is $O(\frac{1}{\sqrt{K}})$. Unfortunately it appears that the scheme in this paper can hold only one parameter (among R, $\frac{M}{N}$, F) bounded by a constant, with the other two vary with K. Other schemes based on (c, d)-caching line graphs could prove to be useful in arriving at coded caching schemes with more interesting and useful parameters.

K	$1 - \frac{M}{N}$	F (this work)	F [2]	R (this work)	R [2]
$\begin{bmatrix} k \\ t \end{bmatrix}_q$	$\frac{\begin{bmatrix} m+t \\ t \end{bmatrix}_q}{\begin{bmatrix} k \\ t \end{bmatrix}_q}$	$\begin{bmatrix} k \\ m+t \end{bmatrix}_q$	$\binom{K}{\frac{KM}{N}}$	$\frac{\begin{bmatrix} m+t \\ t \end{bmatrix}_q}{\begin{bmatrix} k-m \\ t \end{bmatrix}_q}$	$\frac{K(1-\frac{M}{N})}{\frac{MK}{N}+1}$
$(k = 6$ $t = 1)$ 63	$(m = 3)$ $\frac{5}{21}$	651	$\binom{63}{48}$	$\frac{15}{7}$	$\frac{15}{49}$
$(k = 8$ $t = 1)$ 127	$(m = 4)$ $\frac{31}{197}$	2667	$\binom{127}{96}$	$\frac{31}{7}$	$\frac{31}{97}$

TABLE II: For some specific values of $K, 1 - \frac{M}{N}$, we compare the results of [2] with this work.

Acknowledgment

The author would like to thank Dr. Girish Varma for fruitful discussions regarding this work. This work was supported partly by the Early Career Research Award (ECR/2016/000447) from Science and Engineering Research Board (SERB) to Prasad Krishnan.

References

[1] P. Krishnan, "Coded Caching via Line Graphs of Bipartite Graphs", Available on Arxiv at https://arxiv.org/abs/1805.08955.

[2] M.A. Maddah-Ali and U. Niesen, "Fundamental limits of caching", IEEE Transactions on Information Theory, Vol. 60, No. 5, Mar 2014, pp. 2856-2867.

[3] M. Ji, G. Caire, and A. F. Molisch,"Fundamental limits of caching in wireless d2d networks", IEEE Trans. Inform. Theory, vol. 62, no. 2, pp. 849-869, Feb. 2016.

[4] R. Pedarsani, M. A. Maddah-Ali, and U. Niesen, "Online Coded Caching", IEEE/ACM Trans. on Networking, Vol. 24, No. 2, April 2016, pp. 836 - 845.

[5] K. Shanmugam, M. Ji, A. M. Tulino, J. Llorca, and A. G. Dimakis, "Finite-length analysis of caching-aided coded multicasting", IEEE Trans. Inf. Theory, Vol. 62, No. 10, pp. 5524-5537, Oct. 2016.

[6] Q. Yan, X. Tang, Q. Chen, M. Cheng, "Placement Delivery Array Design through Strong Edge Coloring of Bipartite Graphs", IEEE Comm. Letters, Vol. 22, Issue 2, Feb. 2018, pp. 236-239.

[7] Q. Yan, M. Cheng, X. Tang, Q. Chen, "On the Placement Delivery Array Design for Centralized Coded Caching Scheme", IEEE Trans. on Info. Theory, Vol. 63, No. 9, Sep. 2017, pp. 5821-5833.

[8] C. Shangguan, Y. Zhang and G. Ge, "Centralized coded caching schemes: A hypergraph theoretical approach", Available at https://arxiv.org/abs/1608.03989, Aug. 2016.

[9] K. Shanmugam, A. M. Tulino, and A. G. Dimakis, "Coded Caching with Linear Subpacketization is Possible using Ruzsa-Szemredi Graphs", IEEE ISIT 2017, June 25-30, pp. 1237-1241.

[10] L. Tang, A. Ramamoorthy, "Coded Caching Schemes with Reduced Subpacketization from Linear Block Codes", IEEE Trans. on Info. Theory, Vol. 64, No. 4, April 2018, pp. 3099-3120.

[11] M. Cheng, Q. Yan, X. Tang, J. Jiang, "Coded Caching Schemes with Low Rate and Subpacketizations", Available at https://arxiv.org/abs/1703.01548, Oct 2017.

[12] M. Cheng, J. Jiang, Q. Yan, X. Tang, "Coded Caching Schemes for Flexible Memory Sizes", Available at https://arxiv.org/abs/1708.06650, Oct 2017.

[13] R. Diestel, "Graph Theory", Second Edition, Springer-Verlag, 2000.

[14] J. W. P. Hirschfeld, "Projective Geometries over Finite Fields", Second Edition, Oxford Mathematical Monographs, Clarendon Press, 1998.