Characterization and Lower Bounds for Branching Program Size using Projective Dimension

Krishnamoorthy Dinesh*  Sajin Koroth*  Jayalal Sarma*

April 27, 2016

Abstract

We study projective dimension, a graph parameter (denoted by \( pd(G) \) for a graph \( G \)), introduced by Pudlák and Rödl [13], who showed that proving lower bounds for \( pd(G_f) \) for bipartite graphs \( G_f \) associated with a Boolean function \( f \) imply size lower bounds for branching programs computing \( f \). Despite several attempts [13, 17], proving super-linear lower bounds for projective dimension of explicit families of graphs has remained elusive.

We show that there exist a Boolean function \( f \) (on \( n \) bits) for which the gap between the projective dimension and size of the optimal branching program computing \( f \) (denoted by \( \text{bpsize}(f) \)), is \( 2^{\Omega(n)} \). Motivated by the argument in [13], we define two variants of projective dimension - projective dimension with intersection dimension 1 (denoted by \( \text{upd}(G) \)) and bitwise decomposable projective dimension (denoted by \( \text{bitpdim}(G) \)). We show the following results:

(a) There is an explicit family of graphs on \( N = 2^n \) vertices such that
   - the projective dimension is \( O(\sqrt{n}) \).
   - the projective dimension with intersection dimension 1 is \( \Omega(n) \).
   - the bitwise decomposable projective dimension is \( \Omega(n^{1.5}/\log n) \).

(b) We show that there exist a Boolean function \( f \) (on \( n \) bits) for which the gap between \( \text{upd}(G_f) \) and \( \text{bpsize}(f) \) is \( 2^{\Omega(n)} \). In contrast, we also show that the bitwise decomposable projective dimension characterizes size of the branching program up to a polynomial factor. That is, there exists a large constant \( c > 0 \) and for any function \( f \),

\[
\frac{\text{bitpdim}(G_f)}{6} \leq \text{bpsize}(f) \leq (\text{bitpdim}(G_f))^c
\]

(c) We also study two other variants of projective dimension and show that they are exactly equal to well-studied graph parameters - bipartite clique cover number and bipartite partition number respectively. This immediately yields exponential lower bounds for these measures.

*Indian Institute of Technology Madras, Chennai, India. ({kdinesh,sajin,jayalal}@cse.iitm.ac.in)
1 Introduction

For a bipartite graph \( G(U,V,E) \), the projective dimension of \( G \) over a field \( \mathbb{F} \), denoted by \( \text{pd}_\mathbb{F}(G) \), is defined as the smallest \( d \) for which there is a vector space \( W \) of dimension \( d \) (over \( \mathbb{F} \)) and a function \( \phi \) mapping vertices in \( U,V \) to linear subspaces of \( W \) such that for all \((u,v) \in U \times V\), \((u,v) \in E \) if and only if \( \phi(u) \cap \phi(v) \neq \{0\} \). We say that \( \phi \) realizes the graph \( G \).

This graph parameter was introduced by Pudlák and Rödl in [13] where they also showed that for all \( N \), there is a graph (although not explicit) on \( N \) vertices, such that over \( \mathbb{R} \), projective dimension is at least \( \left( \frac{1}{\sqrt{2}} - o(1) \right) \sqrt{N/\log N} \) (see [17]). Later Rónyai, Babai and Ganapathy [17] established the same lower bound over all fields. Pudlák and Rödl [13] showed that for any bipartite graph \( G \) on \( N \) vertices, projective dimension of \( G \) over \( \mathbb{R} \) is at most \( O \left( \frac{N}{\log N} \right) \). Over finite fields, Pudlák and Rödl [13] also showed (by a counting argument) that there exists a graph on \( N \) vertices whose projective dimension is \( \Omega(\sqrt{N}) \). However, till date, obtaining an explicit family of graphs achieving such lower bounds remain elusive. The best lower bound for projective dimension for an explicit graph family is for the bipartite complement of the perfect matchings where a lower bound of \( \epsilon \log N \) for an absolute constant \( \epsilon > 0 \) is known [13] over \( \mathbb{R} \). For a survey on projective dimension and related linear algebraic techniques, refer [14, 10].
For a Boolean function \( f : \{0, 1\}^{2n} \rightarrow \{0, 1\} \), fix a partition of the input bits into two parts of size \( n \) each, and consider the bipartite graph \( G_f(U, V, E) \) defined on vertex sets \( U = \{0, 1\}^n \) and \( V = \{0, 1\}^n \), as \((u, v) \in E\) if and only if \( f(uv) = 1\). We call \( G_f \) as the bipartite realization of \( f \) and to simplify notation, we use \( pd(f) \) to denote \( pd(G_f) \). The original motivation of Pudlák and Rödl [13] for studying projective dimension as a graph parameter was the connection it has to branching program size of the corresponding Boolean function. Pudlák and Rödl [13] showed that if the Boolean function \( f : \{0, 1\}^{2n} \rightarrow \{0, 1\} \) can be computed by a deterministic branching program of size \( s \), then \( pd(f) \leq s \) (for any partition of the input bits into two parts). The proof proceeds by producing a subspace assignment for vertices of \( G_f \) from a branching program computing \( f \). An important consequence of this is that, in order to establish size lower bounds against branching programs, it suffices to prove lower bounds for projective dimension of explicit family of Boolean functions.

**Our Results :** We observe that projective assignment appearing in the proof of [13] also has the property that the dimension of the intersection of two subspaces assigned to the vertices is exactly 1, whenever they intersect. We denote, for a function \( f \), the variant of projective dimension defined by this property as \( upd(f) \) (see Section 4). Clearly, for any Boolean function \( f \), \( pd(f) \leq upd(f) \leq bpsize(f) \). A natural question is whether this restriction helps in proving better lower bounds for the branching programs. We demonstrate that this can help by proving the following theorem.

**Theorem 1.1.** For any \( d \geq 0 \), for the function \( Sl_d \) (on \( 2d^2 \) variables, see Definition 2.3), the projective dimension is exactly equal to \( d \), while the projective dimension with intersection dimension 1 is \( \Omega(d^2) \).

However, this does not directly improve the known branching program size lower bound for \( Sl_d \), since it leads to only a linear lower bound on \( upd(Sl_d) \). We demonstrate the weakness of this measure by showing the existence of a function (although not explicit) for which there is an exponential gap between \( upd \) over any partition and the branching program size (Theorem 5.1). This motivates us to look for variants of projective dimension of graphs, which is closer to the optimal branching program size of the corresponding Boolean function. We observe more properties (see Proposition 2.2) about the subspace assignment from proof of the upper bound from [13]. We call the projective assignments with these properties as bitwise decomposable projective assignment and denote the corresponding dimension\(^1\) as \( bitpdim(f) \) (See Definition 5.2). Thus, for any Boolean function \( f \), \( pd(f) \leq bitpdim(f) \). We also show that \( bitpdim(f) \leq 6 \cdot bpsize(f) \) (Lemma 5.3). To demonstrate the tightness of the definition, we first argue a converse with respect to this new parameter.

**Theorem 1.2.** There is an absolute constant \( c > 0 \) such that if \( bitpdim(f_n) \leq d(n) \) for a function family \( \{f_n\}_{n \geq 0} \) on \( 2n \) bits, then there is a deterministic branching program of size \( (d(n))^c \) computing it.

Thus, super-polynomial size lower bounds for branching programs imply super-polynomial lower bounds for \( bitpdim(f) \). The function \( Sl_d \) (on \( 2d^2 \) input bits - see Definition 2.3) is a natural candidate for proving \( bitpdim \) lower bounds as the corresponding language is hard\(^2\) for the complexity class \( C=L \) under logspace Turing reductions.

---

\(^1\)We do not use the property that intersection dimension is 1 and hence is incomparable with the parameter \( upd \).

\(^2\)Assuming \( C=L \not\subseteq L/poly \), a complete language for \( C=L \) cannot be computed by deterministic branching programs of polynomial size.
However, the best known lower bound for branching program size for an explicit family of functions is $\Omega\left(\frac{n^2}{\log^2 n}\right)$ by Nechiporuk [12] which uses a counting argument on the number of subfunctions. By Theorem 1.2, $\text{bitpdim}(f)$ (for the same explicit function) is at least $\Omega\left(\frac{n^2/c}{\log^2/c n}\right)$. The constant $c$ is large and hence implies only very weak lower bounds for $\text{bitpdim}$. Despite this weak connection, by combining the counting strategy with the linear algebraic structure of $\text{bitpdim}$, we show a super-linear lower bound for $\text{Sl}_d$ matching the branching program size lower bound $^3$

Theorem 1.3 (Main Result). For any $d > 0$, $\text{bitpdim}(\text{Sl}_d)$ is at least $\Omega\left(\frac{d^3}{\log d}\right)$.

Continuing the quest for better lower bounds for projective dimension, we study two further restrictions. In these variants of $\text{pd}$ and $\text{upd}$, the subspaces assigned to the vertices must be spanned by standard basis vectors. We denote the corresponding dimensions as $\text{spd}(f)$ and $\text{uspd}(f)$ respectively. It is easy to see that for any $2n$-bit function, both of these dimensions are upper bounded by $2^n$.

We connect these variants to some of the well-studied graph parameters. The bipartite clique cover number (denoted by $bc(G)$) is the smallest collection of complete bipartite subgraphs of $G$ such that every edge in $G$ is present in some graph in the collection. If we insist that the bipartite graphs in the collection be edge-disjoint, the measure is called bipartite partition number denoted by $bp(G)$. By definition, $bc(G) \leq bp(G)$. These graph parameters are closely connected to communication complexity as well. More precisely, $\log(bp(f))$ is exactly the non-deterministic communication complexity of the function $f$, and $\log(bp(f))$ is a lower bound on the deterministic communication complexity of $f$ (see [7]). In this context, we show the following:

Theorem 1.4. For any Boolean function $f$, $\text{spd}(G_f) = bc(G_f)$ and $\text{uspd}(G_f) = bp(G_f)$.

Thus, if for a function family, the non-deterministic communication complexity is $\Omega(n)$, then we will have $\text{spd}(f) = 2^{\Omega(n)}$. Thus, both $\text{spd}(\text{DISJ})$ and $\text{uspd}(\text{DISJ})$ are $2^{\Omega(n)}$.

2 Preliminaries

In this section, we introduce the notations used in the paper. For definitions of basic complexity classes and computational models, we refer the reader to standard textbooks [7, 18].

Unless otherwise stated we work over the field $\mathbb{F}_2$. We remark that our arguments do generalize to any finite field. All subspaces that we talk about in this work are linear subspaces. Also $\vec{0}$ and $\{0\}$ denotes the zero vector, and zero-dimensional space respectively.

For a graph $G(U, V, E)$, recall the definition of projective dimension of $G$ over a field $\mathbb{F}(\text{pd}_{\mathbb{F}}(G))$, defined in the introduction. For a Boolean function $f : \{0, 1\}^{2n} \to \{0, 1\}$, fix a partition of the input bits into two parts of size $n$ each, and consider the bipartite graph $G_f$ defined on vertex sets $U = \{0, 1\}^n$ and $V = \{0, 1\}^n$, as $(u, v) \in E$ if and only if $f(uv) = 1$. A $\phi$ is said to realize the function $f$ if it realizes $G_f$. Unless otherwise mentioned, the partition is the one specified in the definition of the function.

A deterministic branching program is a directed acyclic graph $G$ with distinct start ($V_0$), accept ($V_+$) and reject ($V_-$) nodes. Accept and reject nodes have fan-out zero and are called sink nodes. Vertices of the DAG, except sink nodes are labeled by variables and have two outgoing edges, one

$^3$A lower bound of $\Omega\left(\frac{d^3}{\log d}\right)$ for the branching program size can also be obtained using Nechiporuk’s method.
labeled 0 and the other labeled 1. For a vertex labeled $x_i$, if input gives it a value $b \in \{0,1\}$, then the edge labeled $b$ incident to $x_i$ is said to be closed and the other edge is open. A branching program is said to accept an input $x$ if and only if there is a path from $V_0$ to $V_+$ along the closed edges in the DAG. A branching program is said to compute an $f : \{0,1\}^n \to \{0,1\}$, if for all $x \in \{0,1\}^n$, $f(x) = 1$ iff branching program accepts $x$. We denote by $\text{bpsize}(f)$ the number of vertices (including accept and reject nodes) in the optimal branching program computing $f$.

**Theorem 2.1** (Pudlák-Rödl Theorem [13]). For a Boolean function $f$ computed by a deterministic branching program of size $s$ and $\mathbb{F}$ being any field, $\text{pd}_{\mathbb{F}}(G_f) \leq s$.

We reproduce the proof of the above theorem in our notation, in Appendix A and derive the following proposition.

**Proposition 2.2.** For a Boolean function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ computed by a deterministic branching program of size $s$, there is a collection of subspaces of $\mathbb{F}^s$ denoted $\mathcal{C} = \{U_i\}_{i \in [n], a \in \{0,1\}}$ and $\mathcal{D} = \{V_j\}_{j \in [n], b \in \{0,1\}}$, where we associate the subspace $U_i^a$ with a bit assignment $x_i = a$ and $V_j^b$ with $y_j = b$ such that if we define the map $\phi$ assigning subspaces from $\mathbb{F}^s$ to vertices of $G_f(U,V,E)$ as $\phi(x) = \text{span}(U_i^a), \phi(y) = \text{span}(V_j^b)$, for $x \in X, y \in Y$ then the following holds true.

- for all $(u,v) \in U \times V$, $\phi(u) \cap \phi(v) \neq \{0\}$ if and only if $f(u,v) = 1$.
- for all $(u,v) \in U \times V$, $\dim(\phi(u) \cap \phi(v)) \leq 1$.
- all the subspaces in $\mathcal{C}, \mathcal{D}$ is equal to the span of some subset of difference of standard basis vectors of $\mathbb{F}^s$.

*Proof.* We reuse the notations introduced in proof of Theorem 2.1 which we have described in the Appendix A. If $H_x$ denotes the set of edges that are closed on an input $a$, then the subspace assignment $\phi(a)$ is span of vectors associated with edges of $H_x$. Denote by $H_{x_i=a_i}$ the subgraph consisting of edges labeled $x_i = a_i$. Hence $H_a$ can be written as span of vectors associated with $H_{x_i=a_i}$. Hence $\phi(a)$ can be expressed as $\text{span}^n U_i$ where $U_i = \text{span}(u,v) \in H_{x_i=a_i}(e_u - e_v)$. A similar argument shows that $\phi(y)$ also has such a decomposition. We now argue the properties of $\phi$.

Note that the first and third property directly follow from proof. To see second property, observe that the branching program is deterministic and hence there can be only one accepting path. Since we observed that the vectors in the accepting path contribute to the intersection space and since there is only one such path, dimension of the intersection spaces is bound to be 1. 

We define the following family of functions and family of graphs based on subspaces of a vector space and their intersections.

**Definition 2.3** ($\text{Sl}_d, \mathcal{P}_d$). Let $\mathbb{F}$ be a finite field. Denote by $\text{Sl}_d$, the Boolean function defined on $\mathbb{F}^{d \times d} \times \mathbb{F}^{d \times d} \to \{0,1\}$ as for any $A, B \in \mathbb{F}^{d \times d}, \text{Sl}_d(A,B) = 1$ if and only if $\text{rowspan}(A) \cap \text{rowspan}(B) \neq \{0\}$. Note that the rowspan is over the field $\mathbb{F}$ (which, in our case, is $\mathbb{F}_2$). Denote by $\mathcal{P}_d$, the bipartite graph $(U,V,E)$ where $U$ and $V$ are the set of all subspaces of $\mathbb{F}^d$. And for any $(I,J) \in U \times V$, $(I,J) \in E \iff I \cap J \neq \{0\}$

We collect the definitions of Boolean functions which we deal with in this work. For $(x,y) \in \{0,1\}^n \times \{0,1\}^n$, $\text{IP}_n(x,y) = \sum_{i=1}^n x_i y_i \mod 2$, $\text{EQ}_n(x,y)$ is 1 if $\forall i \in [n] x_i = y_i$ and is 0 otherwise, $\text{INEQ}_n(x,y) = \neg \text{EQ}_n(x,y)$ and $\text{DISJ}_n(x,y) = 1$ if $\forall i \in [n] x_i \wedge y_i = 0$ and is 0 otherwise. Note that
all the functions discussed so far has branching programs of size \( O(n) \) computing them and hence have projective dimension \( O(n) \) by Theorem 2.1.

Let \( m \in \mathbb{N} \) and \( n = 2m \log m \). The Boolean function, Element Distinctness, denoted \( \text{ED}_n \) is defined on \( 2m \) blocks of \( 2 \log m \) bits, \( x_1, \ldots, x_m \) and \( y_1, \ldots, y_m \) bits and it evaluates to 1 if and only if all the \( x,s \) and \( y,s \) take distinct values when interpreted as integers in \([m^2]\). Let \( q \) be a power of prime congruent to 1 modulo 4. Identify elements in \( \{0,1\}^n \) with elements of \( \mathbb{F}_q^* \). For \( x, y \in \mathbb{F}_q^* \), the Paley function \( \text{PAL}_q^2(x, y) \) is 1 if \( x - y \) is a quadratic residue in \( \mathbb{F}_q^* \) and 0 otherwise.

We observe for any induced subgraph \( H \) of \( G \), if \( G \) is realized in a space of dimension \( d \), then \( H \) can also be realized in a space of dimension \( d \). For any \( d \in \mathbb{N} \), \( P_d \) appears as an induced subgraph of the bipartite realization of \( \text{SI}_d \). Hence, \( \text{pd}(\text{SI}_d) \geq \text{pd}(P_d) \).

**Linear Algebra Basics:** We need the following definition of Gaussian coefficients. For non-negative integers \( n, k \) and a prime power \( q \), \( \left( \begin{array}{c} n \\ k \end{array} \right)_q \) is the expression, \( \frac{(q^n-1)(q^n-q)...(q^n-q^{k-1})}{(q^k-1)(q^k-q)...(q^k-q^{k-1})} \) if \( n \geq k, k \geq 1 \), 0 if \( n < k, k \geq 1 \), 1 if \( n \geq 0, k = 0 \).

We recall some basic lemmas from linear algebra which we use later. Unless otherwise mentioned, all our algebraic formulations are over finite fields \( (\mathbb{F} \text{ of size } q) \). For vector spaces \( V_1, V_2 \) with dimensions \( k_1, k_2 \) respectively, direct sum \( V_1 \oplus V_2 \) is the vector space formed by the column space of the matrix \( M = \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \end{bmatrix} \) where \( B_1 \) is a \( k_1 \times k_1 \) matrix whose column space forms \( V_1 \), \( B_2 \) is a \( k_2 \times k_2 \) matrix whose column space form \( V_2 \). We now state a useful property of direct sum.

**Proposition 2.4.** For an arbitrary field \( \mathbb{F} \), let \( U_1, V_1 \) be subspaces of \( \mathbb{F}^{k_1} \) and \( U_2, V_2 \) be subspaces of \( \mathbb{F}^{k_2} \). Then, \( (U_1 \oplus U_2) \cap (V_1 \oplus V_2) \neq \{0\} \iff U_1 \cap V_1 \neq \{0\} \) or \( U_2 \cap V_2 \neq \{0\} \)

Let \( U, V \) be two vector spaces. Then the vector space formed by \( \text{Span} \{u^Tv \mid u \in U, v \in V\} \) is called the tensor product of vector spaces \( U, V \) denoted as \( U \otimes V \). Here \( u, v \) are column vectors. A basic fact about tensor product that we need is the following : (See Halmos [6] Sec 25). Let \( U \) be a vector space having basis \( u_1, u_2, \ldots, u_k \) and \( V \) be a vector space having basis \( v_1, v_2, \ldots, v_\ell \) over some field \( \mathbb{F} \) then, vector space \( U \otimes V \) has a basis \( B = \{u_i^T v_j \mid i \in \{1, 2, \ldots, k\}, j \in \{1, 2, \ldots, \ell\}\} \) where \( u, v \) are column vectors. Hence, for any two vector spaces \( U, V \), \( \dim(U \otimes V) = \dim(U) \cdot \dim(V) \).

**Proposition 2.5.** For an arbitrary field \( \mathbb{F} \), let \( U_1, V_1 \) be subspaces of \( \mathbb{F}^{k_1} \) and \( U_2, V_2 \) be subspaces of \( \mathbb{F}^{k_2} \). Then, \( (U_1 \oplus U_2) \cap (V_1 \oplus V_2) \neq \{0\} \iff U_1 \cap V_1 \neq \{0\} \) and \( U_2 \cap V_2 \neq \{0\} \)

For a proof of the two Propositions 2.4 and 2.5 refer textbook [16]. Let \( V \) be a finite dimensional vector space. For any \( U \subseteq S V, V = U \oplus U^\perp \). Hence for any \( v \in V \) there exists a unique \( u \in U, w \in U^\perp \) such that \( v = u + w \). A projection map \( \Pi_U \) is a linear map defined as \( \Pi_U(v) = u \) where \( u \) is the component of \( v \) in \( U \). For any \( A, B \subseteq S V \) with \( A \cap B = \{0\} \), let \( V = A + B \). Then any vector \( w \in V \) can be uniquely expressed as \( w = \Pi_A(w) + \Pi_B(w) \). It is easy to see that, for any \( A, B \subseteq \mathbb{F}^d \), with \( A \cap B = \{0\} \), and any \( w \in \mathbb{F}^d \), \( \Pi_{A+B}(w) = \Pi_A(w) + \Pi_B(w) \).

### 3 Properties of Projective Dimension

In this section, we observe properties about projective dimension as a measure of graphs and Boolean functions.
3.1 Bounds for $\text{pd}_F$ on $\land$ and $\lor$ Operations

We start by proving closure properties of projective dimension under Boolean operations $\land$ and $\lor$.

**Lemma 3.1.** Let $F$ be an arbitrary field. For any two functions $f_1 : \{0,1\}^{2n} \rightarrow \{0,1\}$, $f_2 : \{0,1\}^{2n} \rightarrow \{0,1\}$,

- $\text{pd}_F(f_1 \lor f_2) \leq \text{pd}_F(f_1) + \text{pd}_F(f_2)$ and
- $\text{pd}_F(f_1 \land f_2) \leq \text{pd}_F(f_1) \cdot \text{pd}_F(f_2)$

**Proof.** In this proof, for a Boolean $f$ with bipartite representation $G_f(U,V,E)$ we define the map $\phi$ to be from $\{0,1\}^n \times \{0,1\}$ where $\phi(u,0)$ denotes the subspace assigned to $u \in U$ and $\phi(v,1)$ denotes the subspace assigned to $v \in V$ of $G_f$. Let $f_1$ and $f_2$ be of projective dimensions $k_1$ and $k_2$ realized by maps $\phi_1 : \{0,1\}^n \times \{0,1\} \rightarrow F^{k_1}, \phi_2 : \{0,1\}^n \times \{0,1\} \rightarrow F^{k_2}$ respectively.

- From $\phi_1$ and $\phi_2$ we construct a subspace assignment $\phi : \{0,1\}^n \times \{0,1\} \rightarrow F^{k_1+k_2}$ which realize $f_1 \lor f_2$ thus proving the theorem. The subspace assignment is: for $u \in \{0,1\}^n, \phi(u,0) = \phi_1(u,0) \oplus \phi_2(u,0)$. Similarly for $v \in \{0,1\}^n, \phi(v,1) = \phi_1(v,1) \oplus \phi_2(v,1)$. Now, for $u,v \in \{0,1\}^n$, if $f(u, v) = 1$ then it must be that $f_1(u,v) = 1$ or $f_2(u,v) = 1$. Thus either $\phi_1(u,0) \cap \phi_1(v,1) \neq \{0\}$ or $\phi_2(u,0) \cap \phi_2(v,1) \neq \{0\}$. By Proposition 2.4, it must be the case that $(\phi_1(u,0) \oplus \phi_2(u,0)) \cap (\phi_1(v,1) \oplus \phi_2(v,1)) \neq \{0\}$. Hence $\phi(u,0) \cap \phi(v,1) \neq \{0\}$. The dimension of resultant space is $k_1 + k_2$.

- From $\phi_1$ and $\phi_2$ we construct a subspace assignment $\phi : \{0,1\}^n \times \{0,1\} \rightarrow F^{k_1+k_2}$, realizing $f_1 \land f_2$ thus proving the theorem. Consider the following projective dimension assignment $\phi$: for $u \in \{0,1\}^n, \phi(u,0) = \phi_1(u,0) \otimes \phi_2(u,0)$. Similarly for $v \in \{0,1\}^n, \phi(v,1) = \phi_1(v,1) \otimes \phi_2(v,1)$. The proof is similar to the previous case and applying Proposition 2.5 completes the proof.

\[ \square \]

The $\lor$ part of the above lemma was also observed without proof in [14]. We provide a complete proof below. A natural question is whether we can improve any of the above bounds. In that context, we make the following remarks.

**Remark 3.2.**

- We now prove that the construction for $\lor$ tight up to constant factors. Assume $n$ is a multiple of 4. Consider the functions $f(x_1, \ldots, x_n, x_{\frac{n}{4}}, x_{\frac{n}{4}+1}, \ldots, x_n)$ and $g(x_1, x_{\frac{n}{4}+1}, \ldots, x_{\frac{n}{2}}, x_{\frac{n}{2}+1}, \ldots, x_n)$ each of which performs inequality check on the first $\frac{n}{4}$ and the second $\frac{n}{4}$ variables.

  It is easy to see that $f \lor g$ is the inequality function on $\frac{n}{4}$ variables $x_1, \ldots, x_{\frac{n}{4}}$ and the next $\frac{n}{4}$ variables $x_{\frac{n}{2}+1}, \ldots, x_n$. By the fact that they are computed by $n$ size branching programs and using Theorem 2.1 (Pudlák-Rödl theorem) we get that $\text{pd}(f) \leq n$ and $\text{pd}(g) \leq n$. Hence by Lemma 3.1 $\text{pd}(f \lor g) \leq \text{pd}(f) + \text{pd}(g) \leq 2n$. Lower bound on projective dimension of inequality function comes from [13] Theorem 4], giving $\text{pd}(f \lor g) \geq \epsilon \cdot \frac{n}{2}$ for an absolute constant $\epsilon$. This shows that $\text{pd}(f \lor g) = \Theta(n)$.

- A natural idea to improve the upper bound of $\text{pd}(f_1 \land f_2)$ is to prove upper bounds for $\text{pd}(\neg f)$ in terms of $\text{pd}(f)$. However, we remark that over $\mathbb{R}$, it is known [13] that $\text{pd}_R(\text{INEQ}_n)$ is $\Omega(n)$ while $\text{pd}_R(\text{EQ}_n) = 2$. Hence we cannot expect a general relation connecting $\text{pd}_R(f)$ and $\text{pd}_R(\neg f)$.
3.2 A Characterization of Projective Dimension over Finite Fields

We now observe a characterization of bipartite graphs having projective dimension at most \( d \) over \( \mathbb{F} \).

**Lemma 3.3** (Characterization). Suppose \( G \) be a bipartite graph with no two vertices having same neighborhood, \( \text{pd}(G) \leq d \) if and only if \( G \) is an induced subgraph of \( \mathcal{P}_d \).

**Proof.** Suppose \( G \) appears as an induced subgraph of \( \mathcal{P}_d \). To argue, \( \text{pd}(G) \leq d \), simply consider the assignment where the subspaces corresponding to the vertices in \( \mathcal{P}_d \) are assigned to the vertices of \( G \).

On the other hand, suppose \( \text{pd}(G) \leq d \). Let \( U_1, \ldots, U_N \) and \( V_1, \ldots, V_N \) be subspaces assigned to the vertices. Since the neighborhoods of the associated vertices are different, by Proposition 3.4, no two subspaces assigned to these vertices can be the same. Hence corresponding to each vertex in \( G \), there is a unique vertex in \( \mathcal{P}_d \) which corresponds to the assignment. Now the subgraph induced by the vertices corresponding to these subspaces in \( \mathcal{P}_d \) must be isomorphic to \( G \) as the subspace assignment map for \( G \) preserves the edge non-edge relations in \( G \).

An immediate application of this characterization is that \( \text{pd}(\mathcal{P}_d) \leq d \). Observe that vertices with different neighborhoods should be assigned different subspaces. Let \( f: \{0,1\}^n \times \{0,1\}^n \to \{0,1\} \), and \( G_f(X,Y,E) \) be its bipartite realization. Let \( \text{pd}(G_f) = d \).

**Proposition 3.4.** For any subspace assignment \( \phi \) realizing \( G_f \), no two vertices from the same partition whose neighborhoods are different can get the same subspace assignment.

**Proof.** Suppose there exists \( x, x' \in S \) from the same partition, i.e., either \( X \) or \( Y \), such that \( \phi(x) = \phi(x') \). Since \( N(x) \neq N(x') \), without loss of generality, there exists \( z \in N(x) \setminus N(x') \). Now since \( \phi(x) = \phi(x') \), \( x' \) will be made adjacent to \( z \) by the assignment and hence \( \phi \) is no longer a realization of \( G_f \) since \( z \) should not have been adjacent to \( x' \).

For \( \text{pd}(\mathcal{P}_d) \), all vertices on either partitions have distinct neighborhoods. The number of subspaces of a vector space of dimension \( d - 1 \) is strictly smaller than the number of vertices in \( \mathcal{P}_d \). Thus, we conclude the following theorem.

**Theorem 3.5.** For any \( d \in \mathbb{N} \), \( \text{pd}(\mathcal{P}_d) = \text{pd}(\text{Sl}_d) = d \).

For an \( N \) vertex graph \( G \), the number of vertices of distinct neighborhood can at most be \( N \). Thus, the observation that we used to show the lower bound for the graph \( \text{pd}(\mathcal{P}_d) \) cannot be used to obtain more than a \( \sqrt{\log N} \) lower bound for \( \text{pd}(G) \). Also, for many functions, the number of vertices of distinct neighborhood can be smaller. We argue below that by incurring an additive factor of \( 2 \log N \), any graph \( G \) on \( N \) vertices can be transformed into a graph \( G' \) on \( 2N \) vertices such that all the neighborhoods of vertices in one partition are all distinct.

Let \( f: \{0,1\}^{2n} \to \{0,1\} \) be such that the neighborhoods of \( G_f \) are not necessarily distinct. We consider a new function \( f' \) whose bipartite realization will have two copies of \( G_f \) namely \( G_1(A_1, B_1, E_1) \) and \( G_2(A_2, B_2, E_2) \) where \( A_1, A_2, B_1, B_2 \) are disjoint and add a matching connecting vertices in \( A_1, B_2 \) and \( A_2, B_1 \). Since the matching edges associated with every vertex is unique, the neighborhoods of all vertices are bound to be distinct.

Thus the resultant graph has two copies of \( G_f \) with a matching connecting vertices in one partition of a \( G_1 \) to vertices in opposite partition on \( G_2 \). Hence \( G_f' \) can be seen as union of two \( G_f \)
and a matching. Hence applying Lemma 3.1 and observing that matching (or equality function) has projective dimension at most \( n \), \( \text{pd}(f') \leq 2\text{pd}(f) + 2n \). This shows that to show super-linear lower bounds on projective dimension for \( f \) where the neighborhoods may not be distinct, it suffices to show a super-linear lower bound for \( f' \).

4 Projective Dimension with Intersection Dimension 1

Motivated by the proof of Theorem 2.1 (presented in Appendix A) we make the following definition.

**Definition 4.1 (Projective Dimension with Intersection Dimension 1).** A Boolean function \( f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\} \) with the corresponding bipartite graph \( G(U,V,E) \) is said to have projective dimension with intersection dimension 1 (denoted by \( \text{upd}(f) \)) if \( d \) is the smallest possible dimension for which there exists a vector space \( K \) of dimension \( d \) over field \( F \) with a map \( \phi \) assigning subspaces of \( K \) to \( U \cup V \) such that

- for all \((u,v) \in U \times V\), \( \phi(u) \cap \phi(v) \neq \{0\} \) if and only if \((u,v) \in E\).
- for all \((u,v) \in U \times V\), \( \dim (\phi(u) \cap \phi(v)) \leq 1 \).

By the properties observed in Proposition 2.2, Theorem 4.2.

For a Boolean function \( f \) computed by a deterministic branching program of size \( s \), \( \text{upd}_F(f) \leq s \) for any field \( F \).

Thus, it suffices to prove lower bounds for \( \text{upd}(f) \) in order to obtain branching program size lower bounds. We now proceed to show lower bounds on \( \text{upd} \).

4.1 Rank Lower Bound for \( \mathcal{P}_d \)

Our approaches use the fact that adjacency matrix of \( \mathcal{P}_d \) has high rank.

**Lemma 4.3.** Let \( M \) be the bipartite adjacency matrix of \( \mathcal{P}_d \), then \( \text{rank}(M) \geq \left[ \frac{d}{d/2} \right] q \geq q^{d/2} \)

**Proof.** For \( 0 \leq i \leq k \leq d \), and subspace \( I,K \subseteq_s \mathbb{F}_q^d \) with \( \dim(I) = i, \dim(K) = k \), define matrix \( W_{ik} \) over \( \mathbb{R} \) as \( W_{ik}(I,K) = 1 \) if \( i + k \leq d \) and \( I \cap K = \{0\} \) and 0 otherwise. This matrix has dimension \( \left[ \frac{i}{d} \right] q \times \left[ \frac{k}{d} \right] q \).

Consider the submatrix \( M_i \) of \( M \) with rows and columns indexed by subspaces of dimension exactly \( i \). Observe that \( W_{ii} = J - M_i \) where \( J \) is an all ones matrix of appropriate order. These matrices are well-studied (see [5]). Closed form expressions for eigenvalues are computed in [3, 11] and the eigenvalues are known to be non-zero. Hence for \( 0 \leq i \leq d/2 \) the matrix \( W_{ii} \) has rank \( \left[ \frac{i}{d} \right] q \).

Since \( W_{ii} = J - M_i \), \( \text{rank}(M_i) \geq \text{rank}(W_{ii}) - 1 \). This shows that \( \text{rank}(M) \geq \text{rank}(M_i) = \left[ \frac{i}{d} \right] q \) for all \( i \) such that \( 2i \leq d \). Choosing \( i = d/2 \) gives \( \text{rank}(M) \geq \left[ \frac{d}{d/2} \right] q - 1 \geq q^{d/4} - 1 \).
4.2 Lower Bound for \( \text{upd}(\mathcal{P}_d) \) from Intersecting Families of Subspaces

To prove a lower bound on \( \text{upd}(\mathcal{P}_d) \) we define a matrix \( N \) from a projective assignment with intersection dimension 1 for \( \mathcal{P}_d \), such that it is equal to \((q - 1)M\). Let \( D = \text{upd}(\mathcal{P}_d) \). We first show that rank of \( N \) is at most \( 1 + \left[ \frac{d}{1} \right]_q \). Then by Lemma 4.3 we get that rank of \( N \) is at least \( q^{\frac{d^2}{4}} \). Let \( \mathcal{G} = \{G_1, \ldots, G_m\}, \mathcal{H} = \{H_1, \ldots, H_m\} \) be the subspace assignment with intersection dimension 1 realizing \( \mathcal{P}_d \) with dimension \( D \).

**Lemma 4.4.** For any polynomial \( p \) in \( q^x \) of degree \( s \), with matrix \( N \) of order \( |\mathcal{G}| \times |\mathcal{H}| \) defined as \( N[\mathcal{G}_r, \mathcal{H}_t] = p(\dim (\mathcal{G}_r \cap \mathcal{H}_t)) \) with \( \mathcal{G}_r \in \mathcal{G}, \mathcal{H}_t \in \mathcal{H}, \) then \( \text{rank}(N) \leq \sum_{i=0}^{s} \left[ \frac{d}{i} \right]_q \).

**Proof.** This proof is inspired by the proof in [4] of a similar claim where a non-bipartite version of this lemma is proved. To begin with, note that \( p \) is degree \( s \) polynomial in \( q^x \), and hence can be written as a linear combination of polynomials \( p_i = \left[ \frac{r}{i} \right]_q 0 \leq i \leq s \). Let the linear combination be given by \( p(x) = \sum_{i=0}^{s} \alpha_i p_i(x) \). For \( 0 \leq i \leq s \) define a matrix \( N_i \) with rows and columns indexed respectively by \( \mathcal{G}, \mathcal{H} \) defined as \( N_i[\mathcal{G}_r, \mathcal{H}_t] = p_i(\dim (\mathcal{G}_r \cap \mathcal{H}_t)) \). By definition of \( N_i, N = \sum_{i\in[s]} \alpha_i N_i \).

To bound the rank of \( N_i \)'s we introduce the following families of inclusion matrices. For any \( j \in [D] \), consider two matrices \( \Gamma_j \) corresponding to \( \mathcal{G} \) and \( \Delta_j \) corresponding to \( \mathcal{H} \) defined as \( \Gamma_j(G, I) = 1 \) if \( \dim (I) = j, G \in \mathcal{G}, I \subseteq_s G \) and 0 otherwise. \( \Delta_j(H, I) = 1 \) if \( \dim (I) = j, H \in \mathcal{H}, I \subseteq_s H \) and 0 otherwise. Note that rank of the above matrices are upper bounded by the number of columns which is \( \left[ \frac{D}{j} \right]_q \). We claim that for any \( i \in \{0, 1, \ldots, s\} \), \( \text{rank}(N_i) \leq \left[ \frac{D}{i} \right]_q \). This completes the proof since \( N = \sum_{i\in[s]} \alpha_i N_i \).

To prove the claim, let \( \mathcal{F}_i \) denote the set of all \( i \) dimensional subspace of \( \mathbb{F}_q^D \). We show that \( N_i = \Gamma[\mathcal{G}, i] \Delta[\mathcal{H}, i]^T \). Hence \( \text{rank}(N_i) \leq \min \left\{ \text{rank}(\Gamma[\mathcal{G}, i]), \text{rank}(\Delta[\mathcal{H}, i]^T) \right\} \leq \left[ \frac{D}{i} \right]_q \). For \( (G_r, H_t) \in \mathcal{G} \times \mathcal{H}, \Gamma_i(G_r, H_t) = \sum_{I \in \mathcal{F}_i} \Gamma_i(G_r, I) \Delta_i^T (I, H_t) = \sum_{I \in \mathcal{F}_i} \Gamma_i(G_r, I) \Delta_i (H_t, I) = \sum_{I \in \mathcal{F}_i} [I \subseteq_s G_r \cap H_t] = \left[ \dim (G_r \cap H_t) \right]_q = N_i(G_r, H_t) \).

We apply Lemma 4.4 on \( N \) defined via \( p(x) = q^x - 1 \) with \( s = 1 \), to get \( q^{d^2/4} \leq \left[ \frac{d}{d/2} \right]_q \leq 1 + \left[ \frac{D}{q} \right]_q = 1 + (q^D - 1)/(q - 1) \). By definition, \( \text{rank}(N) = \text{rank}(M) \). This gives that \( D = \Omega(d^2) \) and Theorem 1.

4.3 Lower Bound for \( \text{upd}(\mathcal{P}_d) \) from Rectangle Arguments

We now give an alternate proof of for Theorem 1.1 using combinatorial rectangle arguments.

**Lemma 4.5.** For \( f : \{0,1\}^n \rightarrow \{0,1\} \) with \( M_f \) denoting the bipartite adjacency matrix of \( G_f, \text{rank}_{\mathbb{F}} (M_f) \leq q^{O(\text{upd}_p(f))} \) where \( \mathbb{F} \) is a finite field of size \( q \).

**Proof.** Let \( \phi \) be a subspace assignment realizing \( f \) of dimension \( d \) with intersection dimension 1. Let \( S(v) \) for \( v \in \mathbb{F}_q^d \) denote \( \{ (a, b) \in \{0,1\}^n \times \{0,1\}^n \mid \phi(a) \cap \phi(b) = \text{span} \{v\} \} \). Also let \( M_v \) denote the matrix representation of \( S(v) \). That is, \( M_v(a, b) = 1 \iff (a, b) \in S(v) \). Consider all 1 dimensional subspaces which appear as intersection space for some input \((x, y)\). Fix a basis vector for each space and let \( T \) denote the collection of basis vectors of all the intersection spaces. Note that for any \((x, y) \in f^{-1}(1)\), there is a unique \( v \in \mathbb{F}_q^d \) (up to scalar multiples) such that \((x, y) \in S(v) \) for otherwise intersection dimension exceeds 1. Then \( M_f = \sum_{v \in T} M_v \). Now, \( \text{rank}(M_f) \leq \sum_{v \in T} \text{rank}(M_v) \). Since \( \text{rank}(M_v) = 1 \), \( \text{rank}(M_f) \leq |T| \). The fact that the number of 1 dimensional spaces in \( \mathbb{F}_q^d \) can
be at most $2^{d-1}$ completes the proof. We remark that the rank of $M_f$ can be taken over any field (which we choose as $\mathbb{R}$).

We get an immediate corollary. Any function $f$, such that the adjacency matrix of $M_f$ of the bipartite graph $G_f$ is of full rank $2^n$ over some field must have $\text{upd}(f) = \Omega(n)$. There are several Boolean functions with this property, well-studied in the context of communication complexity (see textbook [9]). Hence, we have

**Corollary 4.6.** If the function $f$ is one of $\text{IP}_n$, $\text{EQ}_n$, $\text{INEQ}_n$, $\text{DISJ}_n$ and $\text{PAL}^q_n$, $\text{upd}_F(f)$ is $\Omega(n)$ for any finite field $F$.

For arguing about $\text{PAL}^q_n$, it can be observed that the graph is strongly regular (as $q \equiv 1 \pmod{4}$) and hence the adjacency matrix has full rank over $\mathbb{R}$ [2]. Except for $\text{PAL}^q_n$, all the above functions have $O(n)$ sized deterministic branching programs computing them, Pudlák-Rödl theorem (Theorem 2.1) gives that $\text{upd}$ for these functions (except $\text{PAL}^q_n$) are $O(n)$ and hence the above lower bound is indeed tight.

From Lemma 4.3 it follows that the function $\text{SI}_{d,l}$ also has rank $2^{\Omega(d^2)}$. To see this, it suffices to observe that $\mathcal{P}_d$ appears as an induced subgraph in bipartite realization of $\text{SI}_{d,l}$. Thus, $\text{upd}(\text{SI}_{d,l})$ is $\Omega(d^2)$. We proved in Theorem 3.5 that $\text{pd}(\text{SI}_{d,l}) = d$. This establishes a quadratic gap between the two parameters. This completes the proof of Theorem 1.1.

Let $D(f)$ denote the deterministic communication complexity of the Boolean function $f$. We observe that the rectangle argument used in the proof of Lemma 4.5 is similar to the matrix rank based lower bound arguments for communication complexity. This yields the Proposition 4.7. If $\text{upd}(f) \leq D$, the assignment also gives a partitioning of the 1s in $M_f$ into at most $2^{D-1}$ 1-rectangles. However, it is unclear whether this immediately gives a similar partition of 0s into 0-rectangles as well. Notice that if $D(f) \leq d$, there are at most $2^d$ monochromatic rectangles (counting both 0-rectangles and 1-rectangles) that cover the entire matrix. However, our proof does not exploit this difference.

**Proposition 4.7.** For a Boolean function $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ and a finite field $\mathbb{F}$,

$$\text{upd}_F(f) \leq 2^{D(f)}$$

$$D(f) \leq (\text{pd}_F(f))^2 \log |\mathbb{F}|$$

**Proof.** We give a proof of the first inequality. Any deterministic communication protocol computing $f$ of cost $D(f)$, partitions $M_f$ into $k$ rectangles where $k \leq 2^{D(f)}$ rectangles. Define $f_i : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ for each rectangle $R_i$ $i \in [k]$, such that $f_i(x,y) = 1$ iff $(x,y) \in R_i$. Note that $\text{upd}_F(f_i) = 1$ and $f = \lor_{i=1}^k f_i$. For any $(x,y) \in \{0,1\}^n \times \{0,1\}^n$ if $f(x,y) = 1$, there is exactly one $i \in [k]$ where $f_i(x,y) = 1$. Hence for each $j \in [k]$, $j \neq i$, the intersection vector corresponding to the edge $(x,y)$ in the assignment of $f_j$ is trivial. Hence the assignment obtained by applying Lemma 3.1 to $f_1, \lor f_2 \lor \ldots f_k$ will have the property that for any $(x,y)$ with $f(x,y) = 1$, the intersection dimension is 1. Hence $\text{upd}_F(f) \leq k \leq 2^{D(f)}$. To prove the second inequality, consider the protocol where Alice sends the subspace associated with her input as a $\text{pd}_F(f) \times \text{pd}_F(f)$ matrix.

**Remark 4.8.** We note that the first inequality is tight, up to constant factors in the exponent. To see this, consider the function $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ whose $\text{pd}_F(f) = \Omega(2^n/2)$ [13] Proposition 1 and note that $D(f)$ for any $f$ is at most $n$. Tightness of second inequality is witnessed by $\text{SI}_{d,l}$ since by Lemma 4.3 $D(\text{SI}_{d,l}) = \Omega(d^2)$ while $\text{pd}(\text{SI}_{d,l}) = d$. 11
5 Bitwise Decomposable Projective Dimension

The restriction of intersection dimension being 1, although is a potentially useful for lower bounds for branching program size, does not capture the branching program size exactly. We start the section by demonstrating a function where the gap is exponential.

5.1 Gap between branching program size and upd

In this subsection, we show the existence of a Boolean function $f$ such that the size of the optimal branching program computing it is very high but has a very small projective assignment with intersection dimension 1 for any balanced partition of the input.

**Theorem 5.1.** There exist a function $f : \{0,1\}^n \times \{0,1\}^n$ that requires size $\Omega(2^n)$ for any branching program computing $f$ but the $\text{upd}(f) \leq O(n)$ for any balanced partitioning of the input into two parts.

**Proof.** Consider the function $\text{EQ}_n$. The graph $G_{\text{EQ}_n}(U,V,E)$ with $U = V = N$ is a perfect matching where $N = \{0,1\}^n$. Relabel the vertices in $U$ of this graph to produce a family of $G$ of $N!$ different labeled graphs. Let $F$ be the set of Boolean functions whose corresponding graph is in $G$ (or equivalently $F$ of $N!$ different functions). Let $t$ be smallest number such that any function in $F$ can be computed by a branching program of size at most $t$. The number of branching programs of size $\leq t$ (bounded by $O(t^N)$) forms an upper bound on $|F|$. Thus, $2^{O(t \log t)} \geq N!$, and hence $t$ is $\Omega\left(\frac{2^n}{n}\right)$. Hence there must exist a function $f \in F$ such that $\text{upd}(f) = \text{upd}(\text{EQ}_n) \leq n$ but $\text{bpsize}(f)$ is $\Omega\left(\frac{2^n}{n}\right)$ for this partition.

We now argue upper bound for $\text{upd}(f)$ for any balanced partition. Consider the function $f_\pi$ obtained by a permutation $\pi \in S_N$ on the $U$ part of $\text{EQ}_n$ graph. Consider a partition $\Pi$ of $[2n]$. Let $G_{\text{EQ}_n}^\Pi, G_{f_\pi}^\Pi$ be the corresponding bipartite graphs (and $\text{EQ}_n^\Pi$ and $f_\pi^\Pi$ be the corresponding functions) with respect to the partition $\Pi$, of $\text{EQ}_n$ and $f_\pi$ respectively.

We claim that $\text{upd}(G_{\text{EQ}_n}^\Pi) = \text{upd}(G_{f_\pi}^\Pi)$. By definition for any $(u,v) \in \{0,1\}^n \times \{0,1\}^n$, $f_\pi(u,v) = \text{EQ}_n(\pi^{-1}(u),v)$. Also, let $(u',v')$ be the corresponding inputs according to the partition $\Pi$ of $[2n]$. That is $f_\pi^\Pi(u',v') = f_\pi(u,v) = \text{EQ}_n(\pi^{-1}(u),v)$. Let $x = \pi^{-1}(u)$ and $y = v$. Observe that, for $(x,y) \in \{0,1\}^n \times \{0,1\}^n$ there is unique $(x',y')$ corresponding to it. Hence $f_\pi^\Pi(u',v') = \text{EQ}_n(\pi^{-1}(u),v) = \text{EQ}_n^\Pi(x',y')$. Thus for any input $(u',v')$ of $f_\pi^\Pi$ there is unique input $(x',y')$ of $\text{EQ}_n^\Pi$ obtained via the above procedure. Thus, from the upd assignment for $\text{EQ}_n^\Pi$ we can get a upd assignment for $f_\pi^\Pi$. Observing that Theorem 4.2 holds for any partition $\Pi$ of the input, we get a upd assignment for $\text{EQ}_n^\Pi$.

5.2 A Characterization for Branching Program Size

Motivated by strong properties observed in Proposition 2.2 we make the following definition.

**Definition 5.2 (Bitwise Decomposable Projective Dimension).** Let $f$ be a Boolean function on $2n$ bits and $G_f$ be its bipartite realization. The bipartite graph $G_f(X,Y,E)$ is said to have a bit projective dimension, $\text{bitpd}(G) \leq d$, if there exists a collection of subspaces of $\mathbb{F}_2^d$ denoted $\mathcal{C} = \{U_i^a\}_{i \in [n],a \in \{0,1\}}$ and $\mathcal{D} = \{V_j^b\}_{j \in [n],b \in \{0,1\}}$ where the projective assignment $\phi$ is obtained by associating subspace $U_i^a$ with a bit assignment $x_i = a$ and $V_j^b$ with $y_j = b$ satisfying the conditions listed below.
1. for all \((x, y) \in \{0, 1\}^n \times \{0, 1\}^n\), \(\phi(x) = \text{span}\{U_i^x\}\) and \(\phi(y) = \text{span}\{V_j^y\}\) and \(f\) is realized by \(\phi\).

2. all the subspaces in \(C, D\) is equal to the span of some subset of difference of standard basis vectors of \(\mathbb{F}_2^d\).

3. for any \(S_1, S_2 \subseteq ([n] \times \{0, 1\})\) such that \(S_1 \cap S_2 = \phi\) span \((i,a) \in S_1\) \(U_i^a\) \(\cap\) span \((j,b) \in S_2\) \(V_j^b\) \(=\{0\}\). Same property must hold for subspaces in \(D\).

We show that the new parameter bitwise decomposable projective dimension tightly characterizes the branching program size, up to constants in the exponent.

**Lemma 5.3.** Suppose \(f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}\) has deterministic branching program of size \(s\) then \(\text{bitpdim}(f) \leq 6s\)

**Proof.** The subspace assignment obtained by applying Pudlák-Rödl theorem (Theorem A.1) on an arbitrary branching program need not satisfy Property 3 because there can be a vertex \(z\) that has two edges incident on it reading different variables from the same partition. To avoid this, we subdivide every edge. We show that this transformation is sufficient to get a \(\text{bitpdim}\) assignment. We now give a full proof.

Let \(B\) be a deterministic branching program computing \(f\). Denote the first \(n\) variables of \(f\) as \(x\) and the rest as \(y\). We first apply Pudlák-Rödl transformation on \(B\) to obtain a branching program \(B'\) computing \(f\). We note that \(|V(B')| = |V(B)|\). Obtain \(B''\) from \(B'\) by subdividing every edge \((u, v)\) checking a variable \(x_i = b\) from partition \(x\) to get three edges \((u, V_{uv})\) checking \(x_i = b\) and add two edges between \((V_{uv}, v)\) one which checks \(y_i = 0\) and another which checks \(y_i = 1\).

Clearly the transformation does not change the function computed by the branching program. Since we are taking every edge of the branching program \(B'\) and introducing two more edges, the total number of edges in \(B''\) is \(3|E(B')|\). Since \(B'\) is a deterministic branching program, every vertex \(v \in B'\) has out degree at most 2 and at least 1 for every node except sink node. Hence \(|E(B')| = O(|V(B')|)\). Along with \(|E(B'')| = 3|E(B')|\), we get \(|E(B'')| \leq 6|V(B')| = 6|V(B)|\). Now label every vertex of \(B''\) with standard basis vectors as it is done in Pudlák-Rödl theorem (Theorem A.1). Let \(\phi\) be projective assignment obtained from \(B''\) via Pudlák-Rödl theorem. We claim that \(\phi\) satisfies all the requirements of \(\text{bitpdim}(f)\).

1. Since \(\phi\) is obtained via Pudlák-Rödl it captures adjacencies of \(G_f\). Hence property 1 holds. Property 2 is satisfied by Pudlák-Rödl assignment. (See appendix A)

2. The standard basis vector \(e_u\) corresponding to vertex \(u\) appears only in edges incident on \(u\) in Pudlák-Rödl assignment. For any edge \((u, v)\) querying a variable \(x_i = b\) the other edges incident to \(v\) must query variables from \(y\). All the edges incident on \(u\), except \((u, v)\) must also query variables from \(y\). Otherwise, there is an edge \((w, u)\) which queries a variable \(x_j\) and our transformation would have subdivided the edge. Hence \(e_u, e_v\) belongs only to \(H_{x_i=b}\) amongst \(\{H_{x_i=b}\}_{i\in[n],b\in\{0,1\}}\). This implies Property 3.

We show that given a \(\text{bitpdim}\) assignment for a function \(f\), we can construct a branching program computing \(f\).
**Theorem 5.4** (Theorem 1.2 restated). For a Boolean function \( f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\} \) with \( \text{bitpdim}(f) \leq d \), there exists a deterministic branching program computing \( f \) of size \( d^c \) for some absolute constant \( c \).

**Proof.** Consider the subspace associated with the variables \( C, D \) of the \( \text{bitpdim} \) assignment as the advice string. These can be specified by the a list of \( n \) basis matrices each of size \( d^2 \). Since \( d = \text{bitpdim}(f) = \text{poly}(n) \), the advice string is \( \text{poly}(n) \) sized and only depend on \( n \).

We construct a deterministic branching program computing \( f \) as follows. On input \( x, y \), from the basis matrices in \( C, D \), construct an undirected graph. Note that this is not a deterministic branching program. \( G^* \) with all standard basis vectors in \( C, D \) as vertices and add an edge between two vertices \( u, v \) if \( e_u - e_v \in U_i^\perp \) or \( e_u - e_v \in V_i^{\perp} \) for \( i \in [n] \).

**Lemma 5.5.** On input \( x, y \), \( f(x, y) = 1 \) iff \( G^* \) has a cycle.

**Proof.** Let \( C = C_1 \cup C_2 \) be a cycle in \( G^* \) where \( C_1 \) consists of edges from basis matrices in \( C \) and \( C_2 \) contain edges from basis matrices in \( D \). Note that if one of \( C_1 \) or \( C_2 \) is empty then there is a cycle consisting only of vectors from \( C \) which implies a linear dependence among vectors in \( C \). But this contradicts Property 3 of \( \text{bitpdim} \) assignment. Hence both \( C_1 \) and \( C_2 \) are non-empty.

Then, it must be that \( \sum_{(u,v) \in C_1} e_u - e_v + \sum_{(u,z) \in C_2} e_w - e_z = 0 \). Hence \( \sum_{(u,v) \in C_1} e_u - e_v = -\sum_{(u,z) \in C_2} e_w - e_z \). Hence we get a vector in the intersection which gives \( f(x, y) = 1 \). Note that if \( f(x, y) = 1 \), then clearly there is a non-zero intersection vector. If we express this vector in terms of basis, we get a cycle in \( G^* \).

Hence to check if \( f \) evaluates to 1, it suffices check if there is a cycle in \( G^* \) which is solvable in \( L \) using Reingold’s algorithm [15]. The log-space algorithm can also be converted to an equivalent branching program of size \( n^c \) for a constant \( c \).

Using Theorem 1.2, we demonstrate that the function \( \text{Sl}_d \) is a candidate function (under standard complexity theoretic assumptions) for super-polynomial \( \text{bitpdim} \) lower bounds.

**Proposition 5.6.** The function family \( \{ \text{Sl}_d \}_{d \geq 0} \) is hard for \( C=L \) via logspace Turing reductions.

**Proof.** We start with the following claim.

**Claim 5.7** (Corollary 2.3 of [1]). Fix an \( n \in \mathbb{N} \). There exists a logspace computable function \( g : \mathbb{F}^{n \times n} \to \mathbb{F}^{n \times n} \) such that for any matrix \( M \) over \( \mathbb{F}^{n \times n} \), \( \det(M) = 0 \iff \text{rank}(g(M)) = n \) and \( \det(M) \neq 0 \iff \text{rank}(g(M)) = n - 1 \).

Consider the language \( L = \{ (M_1, M_2) \mid \text{rowspan}(M_1) \cap \text{rowspan}(M_2) \neq \{0\}, M_1, M_2 \in \mathbb{F}^{d \times d} \} \). The reduction is as follows. Given an \( M \in \mathbb{F}^{d \times d} \), apply \( g \) (defined in Claim 5.7) on \( M \) to get \( \tilde{N} \), and define for \( 1 \leq i \leq d \), \( H^i = (M_1^i, M_2^i) \) where \( M_i^j \) is the matrix consisting of the \( i^{th} \) row of \( N \) repeated \( n \) times and \( M_2^j \) as same as \( N \) with \( i^{th} \) row replaced by all 0 vectors. For each \( 1 \leq i \leq d \), we make oracle query to \( L \) checking if \( H^i \in L \) and if all answers are no, reject otherwise accept.

We now argue the correctness of the reduction. Suppose \( \det(M) = 0 \), then \( \tilde{N} = g(M) \) (by Proposition 5.7) must have full rank. Hence for all \( 1 \leq i \leq d \), \( \text{rowspan}(M_1^i) \) and \( \text{rowspan}(M_2^i) \) does not intersect. If \( \det(M) \neq 0 \), then \( \tilde{N} = g(M) \) (by Proposition 5.7) must have a linearly dependent column and hence there is some \( i \) for which \( \text{rowspan}(M_1^i) \) and \( \text{rowspan}(M_2^i) \) is non-zero. Also the overall reduction runs in logspace as \( g \) is logspace computable.

Thus, assuming \( C=L \not\subseteq L/poly \), the function \( \text{Sl}_d \) a complete language for \( C=L \) cannot be computed by deterministic branching programs of polynomial size.

14
5.3 Lower Bounds for Bitwise Decomposable Projective Dimension

From the results of the previous section, it follows that size lower bounds for branching programs do imply lower bounds for bitwise decomposable projective dimension as well. Note that $c$ in Theorem 1.2 is a large constant depending on the space complexity of Reingold’s algorithm [15]. Hence the lower bounds that Theorem 1.2 can give for bitwise decomposable projective dimension are only known to be sub-linear.

To prove super-linear lower bounds for bitwise decomposable projective dimension, we show that Nechiporuk’s method [12] can be adapted to our linear algebraic framework (thus proving Theorem 1.3 from the introduction). The overall idea is the following: given a function $f$ and a bitpdim assignment $\phi$, consider the restriction of $f$ denoted $f_\rho$ where $\rho$ fixes all variables except the ones in $T_i$ to 0 or 1 where $T_i$ is some subset of variables in left partition. For different restrictions $\rho$, we are guaranteed to get at least $c_i(f)$ different functions. We show that for each restriction $\rho$, we can obtain an assignment from $\phi$ realizing $f_\rho$. Hence the number of different bitpdim assignments for $\rho$ restricted to $T_i$ is at least the number of sub functions of $f$ which is at least $c_i(f)$. Let $d_i$ be the ambient dimension of the assignment when restricted to $T_i$. By using the structure of bitpdim assignment, we count the number of assignments possible and use this relation to get a lower bound on $d_i$. Now repeating the argument with disjoint $T_i$, and by observing that the subspace associated with $T_i$s are disjoint, we get a lower bound on $d$ as $d = \sum_i d_i$.

**Theorem 5.8.** For a Boolean function $f: \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ on 22n variables, and $T_1, \ldots, T_m$ are partition of variables to m chunks of size $r_i$ on the first $n$ variables and $c_i(f)$ is the number of distinct sub functions of $f$ when restricted to $T_i$, then $\text{bitpdim}(f) \geq \sum_{i=1}^m \frac{\log c_i(f)}{\log(c_i(f))}$.

**Proof.** Let $(x, y)$ denote the $2n$ input variables of $f$ and $\rho: \{x_1, \ldots, x_n, y_1, \ldots, y_n\} \to \{0,1,*\}$ is a map that leaves only variables in $T_i$ unfixed. Let $\phi$ be a bitpdim assignment realizing $f$ and $G_f(X, Y, Z)$ denote the bipartite realization of $f$. Let $C = \{U_i\}_{i \in [n], \rho(i) \neq *}$ be the associated collection of subspaces. Let $\rho$ be a restriction that does not make $f_\rho$ a constant and $(x,y) \in \{0,1\}^n \times \{0,1\}^n$ which agrees with $\rho$. We use $x, y$ to denote both variables as well as assignment of values. From now on, we fix an $i$ and a partition $T_i$.

Define $L = \text{span}\{U_i^{\rho(i)}\}$ and $R = \text{span}\{V_j^{\rho(n+j)}\}$. For any $x, y \in \{0,1\}^n$ that agrees with $\rho$ on the first $n$ bits, define $Z^x = \text{span}\{U_j^{x_i}\}_{j \in T_i}$ Note that for any $(x, y)$, which agrees with $\rho$, has $\phi(x) = L + Z^x$ and $\phi(y) = R$.

**Property 5.9.** Let $\rho$ be a restriction which does not make the function $f$ constant and which fixes all the variables $y_1, \ldots, y_n$. For all such $\rho$ and $\forall x, y \in \{0,1\}^n$ which agrees with $\rho$, any non-zero $w \in \phi(x) \cap \phi(y)$, where $w = u + v$ with $u \in L$ and $v \in Z^x$ must satisfy $v \neq \emptyset$.

**Proof.** Let there exists an intersection vector $w \in (L + Z^x) \cap R$ with $w = u + v$, $u \in L$ and $v \in Z^x$ and $v \neq \emptyset$. Since $\emptyset \in Z^x$ for any $\hat{x}$, $w = u + \emptyset$ is in $L + Z^x$ and $R$. Thus the function after restriction $\rho$ is a constant. This contradicts the choice of $\rho$. \hfill \square

Note that for any $f_{\rho_1} \neq f_{\rho_2}$, $G_{f_{\rho_1}} \neq G_{f_{\rho_2}}$. Hence the number of bitpdim assignments is at least the number of different number of different sub functions.

We need to give a bitpdim assignment for $G_f(V_1, V_2, E)$ where $V_1 = \{x \mid x$ agrees with $\rho\}$, $V_2 = \{y\}$ where $y = \rho_{[n+1, \ldots, 2n]}$ and $E = \{(x,y) \mid x \in V_1, y \in V_2, f(x,y) = 1\}$. We give an assignment
ψ for $G_\rho$ defined as: $\psi(x) = Z^x$ and $\psi(y) = \text{span}\{\Pi_{Z^x}(R \cap (L + Z^x))\}$. Note that for $(x, y) \in V_1 \times V_2$, $f_\rho(x) = f(x, y)$. Following claim shows that $\psi$ realize $f_\rho$.

**Claim 5.10.** For any $(x, y) \in V_1 \times V_2$, $f(x, y) = 1$ if and only if $\psi(x) \cap \psi(y) \neq \{0\}$.

**Proof.** For any $(x, y) \in X \times Y$, $\phi(x) \cap \phi(y) \neq \{0\}$ if and only if $f(x, y) = 1$. Since $V_1 \subseteq X$ and $V_2 \subseteq Y$, it is enough to prove that for any $(x, y) \in V_1 \times V_2$, $\psi(x) \cap \psi(y) \neq \{0\}$ if and only if $\phi(x) \cap \phi(y) \neq \{0\}$.

We first prove that $\psi(x) \cap \psi(y) \neq \{0\}$ implies $\phi(x) \cap \phi(y) \neq \{0\}$. Let $v$ be a non-zero vector in $\psi(x) \cap \psi(y)$. By definition of $\psi(x)$, there exists a non-empty $J \subseteq V_1$ such that $v = \sum_{\hat{x} \in J} v_{\hat{x}}$ where $v_{\hat{x}} \in Z^\hat{x}$. Also for every $\hat{x} \in J$, there exists a $u_{\hat{x}} \in L$ such that $w_{\hat{x}} = u_{\hat{x}} + v_{\hat{x}}$ and $w_{\hat{x}} \in R$. Define $u$ to be $\sum_{\hat{x} \in J} u_{\hat{x}}$. Since each $w_{\hat{x}}$ is in $L$, $u$ is also in $L$. Hence $w = u + v$ is in $L + Z^x$. Substituting $u$ with $\sum_{\hat{x} \in J} u_{\hat{x}}$ and $v$ with $\sum_{\hat{x} \in J} v_{\hat{x}}$ we get that $w = \sum_{\hat{x} \in J} u_{\hat{x}} + v_{\hat{x}} = \sum_{\hat{x} \in J} w_{\hat{x}}$. Since each $w_{\hat{x}} \in R$, $w \in R$. Hence $w \in R \cap (L + Z^x)$ and $w$ is non-zero as $J$ is non-empty.

Now we prove that $\phi(x) \cap \phi(y) \neq \{0\}$ implies $\psi(x) \cap \psi(y) \neq \{0\}$. Let $w$ be non zero vector in $\phi(x) \cap \phi(y)$ with $w = u + v$ where $u \in L$ and $v \in Z^x$. By Property 5.9 we have $v \neq 0$. By definition $v \in \psi(y)$. Along with $v \in Z^x$, we get $\psi(x) \cap \psi(y) \neq \{0\}$. \qed

Let $Z = \text{span}\{U_{j_1}^0 + U_{j_2}^1 \mid j \in Ti\}$. We now prove that subspace assignment on the only vertex in the right partition of $G_\rho$ which is $\text{span}\{\Pi_{Z^x}(R)\}$ is indeed $\Pi_Z(R)$.

**Claim 5.11.** $\Pi_Z(R) = \text{span}\{\Pi_{Z^x}(R)\}$

**Proof.** We prove that $\text{span}\{\Pi_{Z^x}(R)\} \subseteq \Pi_Z(R)$. Note that $\text{span}\{\Pi_{Z^x}(R)\} = \text{span}\{\text{span}\{\Pi_{Z^x}(w)\}\}$. For an arbitrary $x \in V_1$ and $w \in R$, let $v = \Pi_{Z^x}(w)$. By definition of $Z^x$ and the fact that $\{U_{j_1}^i\}_{i \in [n], j \in \{0, 1\}}$ are disjoint, $\Pi_{Z^x}(w) = +i_{[n], j(i)} = \Pi_{Z^x}(i)(w)$. As $Z = \text{span}\{U_{j_1}^0 + U_{j_2}^1\}$, every $\Pi_{Z^x}(w) \in \Pi_Z(R)$. Hence the span is also in $\Pi_Z(R)$.

Now we show that $\Pi_Z(R) \subseteq \text{span}\{\Pi_{Z^x}(R)\}$. Let $T_i = \{i_1, \ldots, i_k\}$. For $1 \leq j \leq k$ define $x_j^i$ to be $x + e^j$ where $x \in \{0, 1\}^n$ agrees with $\rho$ and for any index $i \in [n]$ with $\rho(i) = \ast$, $x_i = 0$ and $e^j \in \{0, 1\}^n$ is 0 at every index other than $i_j$. Note that for any $j_1 \neq j_2, j_1, j_2 \in Ti$, $Z^{x_{j_1}} \cap Z^{x_{j_2}} = \{0\}$ by Property 3 of Definition 5.2. Also note that $\text{span}\{Z^{x_{j_1}}\} = \text{span}\{U_{j_1}^0\} = Z$. Hence, $\Pi_Z(R) = \text{span}\{\Pi_{Z^x}(R)\}$. But $\text{span}\{\Pi_{Z^x}(R)\} \subseteq \text{span}\{\Pi_{Z^x}(R)\}$. Hence the proof. \qed

For any $\rho$, which fixes all variables outside $T_i$, $Z$ is the same. And since there is only one vertex on the right partition, for different $\rho, \rho'$, $\Pi_Z(R_\rho) = \Pi_Z(R_{\rho'})$ implies $\psi_\rho = \psi_{\rho'}$. Hence to count the number of different $\psi_\rho$'s for different $f_\rho$'s it is enough to count the number of different $\Pi_Z(R)$. To do so, we claim the following property on $\Pi_Z(R)$.

**Property 5.12.** Let $S = \{e_u - e_v | e_u - e_v \in Z\}$. Then there exists a subset $S'$ of $S$ such that all the vectors in $S'$ are linearly independent and $\Pi_Z(R) = \text{span}\{S'\}$. 

16
Proof. By the property of the bitpdim assignment, for any \( i \in [n] \) and \( b \in \{0, 1\}, \ V^b_i = \text{span}\{F^b_i\} \) where \( F^b_i \) is a collection of difference of standard basis vectors. Recall that \( R = \text{span}\{V^\rho_{(n+j)}\} \). Let \( F = \left\{(e_u - e_v) \mid e_u - e_v \in F^\rho_{(n+j)}, j \in [n]\right\} \). Since projections are linear maps and the fact that \( F^\rho_{(n+j)} \) spans \( V^\rho_{(n+j)} \) we get that,

\[
\Pi_Z(R) = \text{span}\{\Pi_Z(w)\}
\]

(1)

Note that \( Z \) is also a span of difference of standard basis vectors. Hence \( \Pi_Z(e_u - e_v) \) is one of \( \bar{0} \), \( e_u - e_w \) or \( e_w - e_v \) where \( e_w \) is some standard basis vector in \( Z \). Let \( S'' = \cup_{e_u - e_v \in F} \Pi_Z(e_u - e_v) \). Hence \( S'' \subseteq S \). Clearly, \( \text{span}\{e_u - e_v\} = \Pi_Z(R) \) by Equation (1). Choose \( S' \) as a linear independent subset of \( S'' \). This completes the proof.

Property 5.12 along with the fact that \( \Pi_Z(R) \) is a subspace of \( Z \), gives us the number of different \( \Pi_Z(R) \) is upper bounded by number of different subsets \( S' \) of \( S \) such that \( |S'| = d_i \) where \( d_i = \text{dim}(Z) \). As \( S' \) is a set of difference of standard basis vectors from \( Z \), \( |S'| \leq \binom{d}{2} \). Thus the number of different such \( S' \) are at most \( \sum_{k=0}^{d} \binom{d}{2} = 2^{O(d_i \log d_i)} \).

Hence the number of restrictions \( \rho \) (that leaves \( T_i \) unfixed) and leading to different \( f_\rho \) is at most \( 2^{O(d_i \log d_i)} \). But the number of such restrictions \( \rho \) is at least \( c_i(f) \). Hence \( 2^{O(d_i \log d_i)} \geq c_i(f) \) giving \( d_i = \Omega \left( \frac{\log c_i(f)}{\log(c_i(f))} \right) \). Using \( d = \sum_i d_i \) completes the proof.

Theorem 5.8 gives a super linear lower bound for Element Distinctness function defined as follows: From an unpublished work of Beame (See [7], Chapter 1), we have \( c_i(ED_1) \geq 2^{n/2}/n \). Hence applying this count to Theorem 5.8 we get that \( d \geq \Omega \left( \frac{n}{\log n} \cdot \frac{n}{\log n} \right) = \Omega \left( \frac{n^2}{\log n} \right) \).

Now we apply this to our context. To get a lower bound using framework described above it is enough to count the number of sub-functions of \( SL_d \).

Lemma 5.13. For any \( i \in [d] \), there are \( 2^\Omega(d^2) \) different restrictions \( \rho \) of \( SL_d \) which fixes all entries other than ith row of the d x d matrix in the left partition.

Proof. Fix any \( i \in [d] \). Let \( S \) be a subspace of \( \mathbb{F}_2^d \). Define \( \rho_S \) to be \( SL_d(A, B) \) where \( B \) is a matrix whose rowspace is \( S \). And \( A \) is the matrix whose all but ith row is 0’s and ith row consists of variables \( (x_{i_1}, \ldots, x_{i_n}) \). Thus for any \( v \in \{0, 1\}^d \), rowspace of \( A(x) \) is exactly \( \text{span}\{v\} \).

We claim that for any \( S, S' \subseteq \mathbb{F}_2^d \) where \( S \neq S' \), \( (SL_d)_{\rho_S} \neq (SL_d)_{\rho_{S'}} \). By definition \( (SL_d)_{\rho_S} \equiv SL_d(A, B) \) and \( (SL_d)_{\rho_{S'}} \equiv SL_d(A', B') \) where \( B \) and \( B' \) are matrices whose rowspaces are \( S \) and \( S' \) respectively. Since \( S \neq S' \) there is at least one vector \( v \in \mathbb{F}_2^d \) such that it belongs to only one of \( S, S' \). Without loss of generality let that subspace be \( S \). Then \( SL_d(A(v), B) = 1 \) as \( v \in S \) where as \( SL_d(A(v), B') = 0 \) as \( v \notin S' \). Hence the number of different restrictions is at least number of different subspaces of \( \mathbb{F}_2^d \) which is \( 2^\Omega(d^2) \). Hence the proof.

This completes the proof of Theorem 1.3 from the introduction. This implies that for \( SL_d \), the branching program size lower bound is \( \Omega \left( \frac{d^2}{\log d} \times d \right) = \Omega \left( \frac{d^3}{\log d} \right) = \Omega \left( \frac{n^{1.5}}{\log n} \right) \) where \( n = 2d^2 \) is the number of input bits of \( SL_d \).
6 Variants of Projective Dimension

In this section, we study three natural variants of projective dimension which are closely related to the projective dimension considered in the earlier sections. Although these do not correspond to the projective assignments related to branching programs, they do shed light on some of the essential features of the projective dimension as a measure for Boolean functions.

We study two stringent variants of projective dimension for which exponential lower bounds and exact characterizations can be derived. Although these measure do not correspond to restrictions on branching programs, they illuminate essential nature of the general measure. We define the measures and show their characterizations in terms of well-studied graph theoretic parameters. We start with the definition of the two variants that we study in this section.

Definition 6.1 (Standard Projective Dimension). A Boolean function \( f : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\} \) with the corresponding bipartite graph \( G(U, V, E) \) is said to have standard projective dimension (denoted by \( \text{spd}(f) \)) over field \( F \), if \( d \) is the smallest possible dimension for which there exists a vector space \( K \) of dimension \( d \) over \( F \) with a map \( \phi \) assigning subspaces of \( K \) to \( U \cup V \) such that

- for all \( (u, v) \in U \times V \), \( \phi(u) \cap \phi(v) \neq \{0\} \) if and only if \( (u, v) \in E \).
- \( u \in U \cup V \), \( \phi(u) \) is spanned by a subset of standard basis vectors in \( K \).

In addition to the above constraints, if the assignment satisfies the property that for all \( (u, v) \in U \times V \), \( \dim(\phi(u) \cap \phi(v)) \leq 1 \), we say that the standard projective dimension is with intersection dimension 1, denoted by \( \text{uspd}(f) \). We make some easy observations about the definition itself.

Remark 6.2. • For \( N \times N \) bipartite graph \( G \) with \( m \) edges, consider the assignment of standard basis vectors to each of the edges and for any \( u \in U \cup V \), \( \phi(u) \) is the span of the basis vectors assigned to the edges incident on \( u \). Moreover, the intersection dimension in this case is 1. Hence for any \( G \), \( \text{spd}(G) \leq \text{uspd}(G) \leq m \).

• Given a subspace assignment realizing \( G_f \) of dimension \( d = \text{pd}_F(f) \) for each non-zero vector in \( F^d \), consider the graph \( G_v \) whose edges have intersection space containing the vector \( v \). This collection covers all edges in \( G_f \) giving \( bc(G_f) \) is upper bounded by number of one dimensional subspaces in \( F^d \). By previous claim \( \text{spd}(f) \leq \frac{\text{pd}_F(f) - 1}{q - 1} \).

Note that for the case of projective dimension even though \( \text{pd}(G) \leq \text{spd}(G) \), there are graphs for which the gap is exponential. For example, consider the bipartite realization \( G \) of \( \text{EQ}_n \) with \( N = 2^n \). We know \( \text{pd}(G) = \Theta(\log N) \) but \( \text{spd}(G) \geq N \) since each of the vertices associated with the matched edges cannot share any basis vector with vertices in other matched edges. Hence dimension must be at least \( N \). We show that standard projective dimension of bipartite \( G \) is same as that of biclique cover number.

Theorem 6.3 (Restatement of Theorem 1.4). For any Boolean function \( f \), \( \text{spd}(G_f) = bc(G_f) \) and \( \text{uspd}(G_f) = bp(G_f) \).

Proof. \((\text{spd}(f) \leq bc(G_f)) \) Let \( G = G_f \), \( t = bc(G) \) and \( A_1, \ldots, A_t \) be a bipartite cover for \( G \). For a vertex \( v \in V(G) \), let \( I_v = \{ e_i \mid v \in A_i \} \). We claim that \( \{ I_v \}_{v \in V(G)} \) is a valid standard projective assignment. Suppose \( I_u \cap I_v \neq \emptyset \), then there exists an \( i \) such that \( u, v \in A_i \) and \( (u, v) \in E(A_i) \). Hence \( (u, v) \in E(G) \). Also if \( (u, v) \in E(G) \), then \( \exists i \) s.t. \( (u, v) \in E(A_i) \). By definition of \( I_u, I_v \), \( e_i \in I_u \cap I_v \) giving \( I_u \cap I_v \neq \emptyset \).
(bc(G_f) \leq spd(G_f)) Let G = G_f, t = spd(G) and \{I_u\}_{u \in V(G)} be the subsets assigned. Consider G_i = \{(u, v) \mid i \in I_u \text{ and } i \in I_v\} for i \in \{1, \ldots, t\}. We claim that the collection of G_i forms a valid bipartite cover of G. If (u, v) \in E(G), we have I_u \cap I_v \neq \emptyset. Hence there exists an i \in I_u \cap I_v and (u, v) \in E(G_i). If (u, v) \in E(G_i) for some i, then i \in I_u \text{ and } i \in I_v \text{ implying } I_u \cap I_v \neq \emptyset. This gives that (u, v) \in E(G) from the definition of standard assignment.

(bp(G_f) \leq uspd(G_f)) Let \phi be the intersection dimension one standard assignment of ambient dimension d of f. For every e_i \in F^d, define the set C_i = \{(x, y) \mid \phi(x, y) = e_i\}. We claim that C_i \subseteq I \times I is a bipartite partition of G_f. Every C_i is a biclique, because if \phi(x, y) = e_i \text{ then that implies } e_i \in \phi(x) \text{ and } e_i \in \phi(y). Note that for every (x, y) \in G_f, there exists a unique i \in [d] such that (x, y) \in C_i. Hence any (x, y) \in G_f belongs to exactly one of the sets C_i thus implying that C_i's are edge disjoint biclique covers. Note that any (x, y) \notin G_f do not belong to any of C_i's as \phi(x, y) = \{0\}.

(uspd(G_f) \leq bp(G_f)) Let C = \{C_i\}_{i \in [d]} where d = bp(G_f) be a biclique partition cover. We give a standard assignment \phi for G_f defined as follows. For any x, \phi(x) = \text{span} \{e_i \mid \exists y, (x, y) \in C_i\}. By definition \phi is a standard assignment. We just need to prove that (x, y) \in G_f if and only \phi(x, y) \neq \{0\} and dim \phi(x, y) = 1. To prove this we would once again employ the rectangle property of bicliques, that is if (x, y') \in C_i and (x', y) \in C_i then so is (x, y'). First we will argue that if there an intersection then it is dimension 1. Recall that intersection of two standard subspaces is a standard subspace. Suppose there exists (x, y) with \text{dim } \phi(x, y) > 1. Let e_j, e_k be any two standard intersection vectors in \phi(x, y). By construction and rectangle property of bicliques, we get that (x, y) \in C_j and (x, y) \in C_k contradicting the disjoint cover property. Hence for any (x, y), \text{dim } \phi(x, y) \leq 1. If (x, y) \notin G_f, then there does not exist an i, (x, y) \in C_i. But if \phi(x, y) = e_i \text{ for some } i \in [d], then that implies by rectangle property of bicliques that (x, y) \in C_i, a contradiction.

7 Conclusion

In this paper we studied variants of projective dimension of graphs with improved connection to branching programs. We showed lower bounds for these measures indicating the weakness and of each of the variants. A pictorial representation of all parameters is as follows.

There are several research threads arising from our work.

Amplifying the gap between \text{upd}(P_d) and \text{pd}(P_d) : We believe that the \Omega(d^2) lower bound on \text{upd}(P_d) is not tight. It is natural to study composition of functions to improve this gap. Because \text{pd} assignments are less stringent and easier to compose where as \text{upd} assignments are more stringent hence are believed to be harder to compose.
Zero-covers of the Communication Matrix and upd: Any upd assignment for a function \( f : \{0,1\}^n \times \{0,1\}^n \) induces a 1-cover of the associated communication matrix \( M_f \) in the following sense. For any vector \( a \) in the ambient space of upd assignment \( \phi \) for \( f \), consider the set \( S_a = \{ (x,y) | \phi(x) \cap \phi(y) = a \} \). It is easy to see that \( S_a \) forms a combinatorial rectangle. Since \( \phi \) is a upd assignment the collection of rectangles \( \{ S_a \}_{a \in \phi} \) forms a partition of 1's in \( M_f \) into 1 rectangles. Thus for any function \( f \), there is a partition of \( M_f \) into \( 2^{\text{upd}(f)} \) many 1 rectangles. But there is no such natural partition for 0 rectangles in \( M_f \) evident from a upd assignment. This highlights a fundamental difference between deterministic communication complexity and uni-dimensional projective assignments. Hence one can try to leverage this potential fundamental difference to prove upd lower bounds which are much better than the deterministic communication complexity lower bounds. This is also an approach to improve the gap between \( \text{upd}(\mathcal{P}_d) \) and \( \text{pd}(\mathcal{P}_d) \).

Improving the Counting Arguments: The subspace counting based bitpdim lower bounds we proved are tight for functions like ED\(_n\). But we also showed that under standard complexity theoretic assumptions the bitpdim assignment for \( \mathcal{P}_d \) is not tight. Hence the specific linear algebraic properties of \( \mathcal{P}_d \) can be used to improve the bitpdim lower bound we obtained for \( \mathcal{P}_d \).

Intersecting Families of Subspaces: Our first lower bound for \( \text{upd}(\mathcal{P}_d) \) using intersection family of subspaces, can be improved if one can improve the upper bound on \( \min \{ \text{rank} (\Gamma[\mathcal{G},i]) , \text{rank} (\Delta[\mathcal{H},i]^T) \} \) to something better than the naive \( \left[ \frac{D}{1} \right]_q \). This can potentially be achieved by proving linear algebraic properties of intersection vectors in \( \text{upd}(\mathcal{P}_d) \). We also note that there is no such direct approach for improving the second lower bound using rectangle arguments.

References


A Proof of Pudlák-Rödl theorem

In this section, we reproduce the proof of the projective dimension upper bound in terms of branching program size. The proof is originally due to [13], but we supply the details which are essential for the observations that we make.

**Theorem A.1.** Let \( f : \{0,1\}^{2n} \rightarrow \{0,1\} \) be computed by a branching program \( \mathcal{B} \) of size \( s \). Let \( G_f \) be the bipartite realization of \( f \), with respect to any partition of \( [2n] \) into two parts and \( \mathbb{F} \) be any arbitrary field. Then, \( \text{pd}(G_f) \leq s \)

**Proof.** It suffices to come up with a subspace assignment \( \phi \) such that \( G_f(P,Q,E) \) has a projective representation in \( \mathbb{F} \). Associate \( u,v \) to be vertices in \( P,Q \) respectively. In other words, \( u \) corresponds to input variables \( \{x_1,x_2,\ldots,x_n\} \) and \( v \) corresponds to \( \{x_{n+1},\ldots,x_{2n}\} \) (corresponding to the given partition). By the acceptance property of branching program \( \mathcal{B} \), \( f(u \circ v) = 1 \iff \exists \) a path from \( V_0 \) to accept in \( \mathcal{B} \). Since vertices in \( G_f \) corresponds to strings in \( \{0,1\}^n \), it suffices to give an assignment \( \phi \) such that

\[ \exists \text{ a path from start to accept in } \mathcal{B} \iff \text{Basis of } \phi(u), \phi(v) \text{ are linearly dependent} \quad (2) \]
We first assign vectors to vertices of the branching program and then use it to come up with a subspace assignment.

Suppose there is a path from \( v_0 \) to accept in \( B \). A simple possible way to have dependence is to have sum of the vectors assigned to the edges of the path telescoping to zero. This can be achieved in the following way.

1. Modify \( B \) by adding a new start vertex labelled with a variable from the other partition from which \( v_0 \) got its label. For example, if \( V_0 \) is labelled with any of \( x_1, x_2, \ldots, x_n \), the new vertex gets its label from \( \{x_{n+1}, \ldots, x_{2n}\} \) and vice-versa. Connect both outgoing edges labelled 0, 1 to \( V_0 \).

2. Merge the accept node with the new start node. Let \( C \) be the resultant graph which is no longer acyclic. Assign standard basis vectors to each vertex in \( C \).

3. Assign to each edge \((u, v)\) the vector \( e_u - e_v \).

Now, the subspace assignment to a vertex \( v \in V(G_f) \) is to take span of all vectors assigned to closed edges\(^4\) on the input \( v \). If there are no closed edges, we assign the zero subspace. With the above modification, cycles in the graph would lead to telescoping of difference vectors (along the cycle edges) to sum to zero.

Modification (1) is necessary as it is possible to have a cycle that does not contain any vertex labelled with \( \{x_{n+1}, \ldots, x_{2n}\} \). Then \( \phi(v) \) will be just zero subspace and \( \phi(u) \cap \phi(v) \) will be trivial when there is a cycle. It is to avoid this that we add a vertex labelled with variable from the other partition.

To show that \( \phi \) is a valid subspace assignment, it remains to show that reverse implication of statement \( \ref{reverse} \) holds. Suppose for \((u, v) \in E(G_f), \phi(u), \phi(v) \) are linearly dependent. Hence there exists a non trivial combination giving a zero sum.

\[
\sum_{e \in E(C), e=(u,v)} \lambda_e (e_u - e_v) = 0, \quad \lambda_e \in \mathbb{F} \forall e \in E(C)
\]

Let \( S \) be the non-empty set of edges such that \( \lambda_e \neq 0 \) and \( V(S) \) be its set of vertices. Now for any vertex \( u \in V(S) \) there must be at least two edges containing \( u \) because with just a single edge \( e_u \), which being a basis vector and summing up to zero, must have a zero coefficient which contradicts that fact that \( e \in S \). This shows that every vertex in \( S \) has a degree \( \geq 2 \) (in the undirected sense). Hence it must have an undirected cycle.

\[\square\]

## B Bounds on the Gaussian Coefficients

**Proposition B.1** (Lemma 1 of [8]). For integers, \( k \geq 0, n \geq k \).

\[ q^{k(n-k)} \leq \binom{n}{k}_q < c_q q^{k(n-k)} \tag{3} \]

where \( c_q = \prod_{j=1}^{\infty} \frac{1}{1-q^{-j}} \). Note that for all \( q \geq 2 \), \( c_q \leq c_2 = 3.462 \ldots \)

\(^4\)Recalling from introduction, a vertex labelled \( x_i \), if the input gives it a value \( b \in \{0, 1\} \), then the edge labelled \( b \) incident to \( x_i \) is said to be “closed” and the other edge is said to be “open”.

22
Proof. Note that since $n \geq k$, $q^n \geq q^k$, we have $\frac{q^{n-t}}{q^k-t} \geq \frac{q^n}{q^k}$ for any $0 \leq t < q^k$. Hence the lower bound follows.

For the upper bound,

$$\binom{n}{k}_q = \frac{(q^n - 1)(q^n - q) \ldots (q^n - q^{k-1})}{(q^k - 1)(q^k - q) \ldots (q^k - q^{k-1})}$$

$$= \frac{q^{nk}}{q^k} \left[ \frac{(1 - q^{-n})(1 - q^{-(n-1)}) \ldots (1 - q^{-(n-k+1)})}{(1 - q^{-1})(1 - q^{-2}) \ldots (1 - q^{-(k-1)})} \right]$$

Numerator of the previous expression can be upper bounded by $q^{nk}$ while denominator can be lower bounded by $q^{k^2}(c_q)^{-1}$. This completes the proof.

Remark B.2. This shows that the total number of subspaces of an $n$ dimensional space is upper bounded by $2c_q \sum_{i=0}^{n/2} q^{in} \leq 2c_2 q^{(n^2+n)/2}$.