Characterization and Lower Bounds for Branching Program Size using Projective Dimension

Krishnamoorthy Dinesh[*] Sajin Koroth[*] Jayalal Sarma[*]

February 9, 2017

Abstract

We study projective dimension, a graph parameter (denoted by $\mathsf{pd}(G)$ for a graph G), introduced by Pudlák and Rödl (1992). For a Boolean function f(on n bits), Pudlák and Rödl associated a bipartite graph G_f and showed that size of the optimal branching program computing f (denoted by $\mathsf{bpsize}(f)$) is at least $\mathsf{pd}(G_f)$ (also denoted by $\mathsf{pd}(f)$). Hence, proving lower bounds for $\mathsf{pd}(f)$ imply lower bounds for $\mathsf{bpsize}(f)$. Despite several attempts (Pudlák and Rödl (1992), Rónyai et.al, (2000)), proving super-linear lower bounds for projective dimension of explicit families of graphs has remained elusive.

We observe that there exist a Boolean function f for which the gap between the $\mathsf{pd}(f)$ and $\mathsf{bpsize}(f)$) is $2^{\Omega(n)}$. Motivated by the argument in Pudlák and Rödl (1992), we define two variants of projective dimension - *projective dimension with intersection dimension 1* (denoted by $\mathsf{upd}(f)$) and *bitwise decomposable projective dimension* (denoted by $\mathsf{bpdim}(f)$). We show the following results :

(a) We observe that there exist a Boolean function f for which the gap between $\mathsf{upd}(f)$ and $\mathsf{bpsize}(f)$ 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 constant $c > 0$ and for any function f,

$$\mathsf{bpdim}(f)/6 \leq \mathsf{bpsize}(f) \leq (\mathsf{bpdim}(f))^c$$

(b) We introduce a new candidate function family f for showing super-polynomial lower bounds for $\mathsf{bpdim}(f)$. As our main result, we demonstrate gaps between $\mathsf{pd}(f)$ and the above two new measures for f :

$$\mathsf{pd}(f) = O(\sqrt{n}) \qquad \mathsf{upd}(f) = \Omega(n) \qquad \mathsf{bpdim}(f) = \Omega\left(\frac{n^{1.5}}{\log n}\right)$$

(c) Although not related to branching program lower bounds, we derive exponential lower bounds for two restricted variants of $\mathsf{pd}(f)$ and $\mathsf{upd}(f)$ respectively by observing that they are exactly equal to well-studied graph parameters - bipartite clique cover number and bipartite partition number respectively.

1 Introduction

A central question in complexity theory - the P vs L problem - asks if a deterministic Turing machine that runs in polynomial time can accept any language that cannot be accepted by deterministic

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

ISSN 1433-8092

Turing machines with logarithmic space bound. A stronger version of the problem asks if P is separate from $\mathsf{L/poly}$ (deterministic logarithmic space given polynomial sized advice). The latter, recast in the language of circuit complexity theory, asks if there exists an *explicit* family of functions ($f : \{0,1\}^n \to \{0,1\}$) computable in polynomial time (in terms of n), such that any family of deterministic branching programs computing them has to be of size $2^{\Omega(n)}$. However, the best known non-trivial size lower bound against deterministic branching programs, due to Nechiporuk [15] in 1970s, is $\Omega(\frac{n^2}{\log^2 n})$.

Pudlák and Rödl [16] described a linear algebraic approach to show size lower bounds against deterministic branching programs. They introduced a linear algebraic parameter called *projective dimension* (denoted by $\mathsf{pd}_{\mathbb{F}}(f)$, over a field $\mathbb{F}$) defined on a natural graph associated with the Boolean function f. 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 $G_f(U, V, E)$ 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. For a bipartite graph $G(U, V, E)$, the projective dimension of G over a field $\mathbb{F}$, denoted by $\mathsf{pd}_{\mathbb{F}}(G)$, is defined as the smallest d for which there is a vector space W of dimension d (over $\mathbb{F}$) and a function ϕ 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 ϕ *realizes* the graph G.

Pudlák and Rödl [16] showed that if f can be computed by a deterministic branching program of size s, then $\mathsf{pd}_{\mathbb{F}}(f) \leq s$ over any field $\mathbb{F}$. Thus, 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.

Pudlák and Rödl in [16] showed that for most Boolean functions $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, $\mathsf{pd}_{\mathbb{R}}(f)$ is $\Omega(\sqrt{\frac{2^n}{n}})$. In a subsequent work, the same authors [17] also established an upper bound $\mathsf{pd}_{\mathbb{R}}(f) = O(\frac{2^n}{n})$ for all functions. More recently, Rónyai, Babai and Ganapathy [20] established the same lower bound over all fields. Over finite fields $\mathbb{F}$, Pudlák and Rödl [16] also showed (by a counting argument) that there exists a Boolean function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ such that $\mathsf{pd}_{\mathbb{F}}(f)$ is $\Omega(\sqrt{2^n})$. However, till date, obtaining an explicit family of Boolean functions (equivalently graphs) achieving such lower bounds remain elusive. The best lower bound for projective dimension for an explicit family of functions is for the inequality function (on $2n$ bits, the graph is the bipartite complement of the perfect matching) where a lower bound of ϵn for an absolute constant $\epsilon > 0$ is known [16] over $\mathbb{R}$. For a survey on projective dimension and related linear algebraic techniques, refer [17, 12]. Thus, the best known size lower bound that was achieved using this framework is only $\Omega(n)$ which is not better than trivial lower bounds.

Our Results : The starting point of our investigation is the observation that projective assignment appearing in the proof of [16] also has the property that the dimension of the intersection of two subspaces assigned to the vertices is exactly 1, whenever they intersect (See Proposition 2.2(2)). We denote, for a function f, the variant of projective dimension defined by this property as $\mathsf{upd}(f)$ (See Section 4). From the above discussion, for any Boolean function f, $\mathsf{pd}(f) \leq \mathsf{upd}(f) \leq \mathsf{bpsize}(f)$. A natural question is whether this restriction helps in proving better lower bounds for the branching programs. By observing properties about projective dimension and choosing a new candidate function[1], we demonstrate that the above restriction can help by proving the following quadratic gap between the two measures.

[1] the candidate function is in NC^2 but unlikely to be in NL. See Proposition 5.5.

Theorem 1.1. *For any $d \geq 0$, for the function SI_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 SI_d, since it leads to only a linear lower bound on $\mathsf{upd}(\mathsf{SI}_d)$. 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 (Proposition 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 the proof of the upper bound from [16]. We call the projective assignments with these properties *bitwise decomposable projective assignment* and denote the corresponding dimension[2] as $\mathsf{bitpdim}(f)$ (See Definition 5.2). Thus, for any Boolean function f, $\mathsf{pd}(f) \leq \mathsf{bitpdim}(f)$. We also show that $\mathsf{bitpdim}(f) \leq 6 \cdot \mathsf{bpsize}(f)$ (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 $\mathsf{bitpdim}(f_n) \leq d(n)$ 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 $\mathsf{bitpdim}(f)$. The function SI_d (on $2d^2$ input bits - See Definition 2.3) is a natural candidate for proving $\mathsf{bitpdim}$ lower bounds as the corresponding language is hard[3] for the complexity class $\mathsf{C_{=}L}$ under logspace Turing reductions.

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 [15] which uses a counting argument on the number of sub-functions. By Theorem 1.2 , $\mathsf{bitpdim}(f)$ (for the same explicit function) is at least $\Omega\left(\frac{n^{2/c}}{\log^{2/c} n}\right)$. The constant c is more[4] than 3 and hence implies only weak lower bounds for $\mathsf{bitpdim}$. Despite this weak connection, by combining the counting strategy with the linear algebraic structure of $\mathsf{bitpdim}$, we show a super-linear lower bound for SI_d matching the branching program size lower bound[5].

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

Theorems 1.1 and 1.3 demonstrate gaps between the pd and the new measures considered. In particular, for $n = d^2$, $\mathsf{pd}(\mathsf{SI}_d) = O(\sqrt{n})$, $\mathsf{upd}(\mathsf{SI}_d) = \Omega(n)$, and $\mathsf{bitpdim}(\mathsf{SI}_d) = \Omega\left(\frac{n^{1.5}}{\log n}\right)$. We remark that Theorem 1.3 implies a size lower bound of $\Omega(\frac{n^{1.5}}{\log n})$ for branching programs computing the function SI_d (where $n = d^2$). However, note that this can also be derived from Nechiporuk's method. For the Element Distinctness function, the above linear algebraic adaptation of Nechiporuk's method for $\mathsf{bitpdim}$ gives $\Omega(\frac{n^2}{\log^2 n})$ lower bounds (for $\mathsf{bitpdim}$ and hence for bpsize) which matches with the best lower bound that Nechiporuk's method can derive. This shows that our modification of approach in [16] can also achieve the best known lower bounds for branching program size.

[2] We do not use the property that intersection dimension is 1 and hence is incomparable with upd.

[3] Assuming $\mathsf{C_{=}L} \not\subseteq \mathsf{L/poly}$, SI_d cannot be computed by deterministic branching programs of size $\mathsf{poly}(d)$.

[4] The value of c can be shown to be at most $3 + \epsilon$. See proof of Theorem 1.2 in Section 5.1.

[5] 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.

Continuing the quest for better lower bounds for projective dimension, we study two further restrictions. In these variants of pd and upd, the subspaces assigned to the vertices must be spanned by standard basis vectors. We denote the corresponding dimensions as $\mathsf{spd}(f)$ and $\mathsf{uspd}(f)$ 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 $\mathsf{bc}(G)$) 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 $\mathsf{bp}(G)$. By definition, $\mathsf{bc}(G) \leq \mathsf{bp}(G)$. These graph parameters are closely connected to communication complexity as well. More precisely, $\log(\mathsf{bc}(G_f))$ is exactly the non-deterministic communication complexity of the function f, and $\log(\mathsf{bp}(G_f))$ is a lower bound on the deterministic communication complexity of f (See [9]). In this context, we show the following:

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

Thus, if for a function family, the non-deterministic communication complexity is $\Omega(n)$, then we will have $\mathsf{spd}(f) = 2^{\Omega(n)}$. Thus, both $\mathsf{spd}(\mathsf{DISJ})$ and $\mathsf{uspd}(\mathsf{DISJ})$ 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 [9, 21].

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 $\{0\}$ denote the zero vector, and zero-dimensional space respectively. For a subspace $U \subseteq \mathbb{F}^n$, we call the ambient dimension of U as n. We denote $e_i \in \mathbb{F}^n$ as the i^{th} standard basis vector with i^{th} entry being 1 and rest of the entires being zero.

For a graph $G(U, V, E)$, recall the definition of projective dimension of G over a field $\mathbb{F}(\mathsf{pd}_{\mathbb{F}}(G))$, defined in the introduction. 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 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 ϕ 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. We denote by $\mathsf{bpsize}(f)$ the number of vertices (including accept and reject nodes) in the optimal branching program computing f.

Theorem 2.1 (Pudlák-Rödl Theorem [16])**.** *For a Boolean function f computed by a deterministic branching program of size s and $\mathbb{F}$ being any field, $\mathsf{pd}_{\mathbb{F}}(G_f) \leq s$.*

The proof of this result proceeds by producing a subspace assignment for vertices of G_f from a branching program computing f. We reproduce the proof of the above theorem in our notation, in Appendix A and derive the following proposition from the same.

Proposition 2.2. *For a Boolean function $f : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ computed by a deterministic branching program of size s, there is a collection of subspaces of $\mathbb{F}^s$ denoted $\mathcal{C} = \{U_i^a\}_{i \in [n], a \in \{0,1\}}$ and $\mathcal{D} = \{V_j^b\}_{j \in [n], b \in \{0,1\}}$, 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 ϕ assigning subspaces from $\mathbb{F}^s$ to vertices of $G_f(U, V, E)$*

as $\phi(x) = \underset{1\le i\le n}{\mathsf{span}}\{U_i^{x_i}\}$, $\phi(y) = \underset{1\le j\le n}{\mathsf{span}}\{V_j^{y_j}\}$, *for* $x \in X, y \in Y$ *then the following holds true. Let* $S = \{e_i - e_j \mid i, j \in [s], i \neq j\}$.

1. *for all* $(u, v) \in U \times V$, $\phi(u) \cap \phi(v) \neq \{0\}$ *if and only if* $f(u, v) = 1$.
2. *for all* $(u, v) \in U \times V$, $\mathsf{dim}\,(\phi(u) \cap \phi(v)) \le 1$.
3. *For any* $W \in \mathcal{C} \cup \mathcal{D}$, $\exists S' \subseteq S$ *such that* $W = \mathsf{span}\,\{S'\}$.

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 $span_{i=1}^n U_i$ where $U_i = span_{(u,v)\in H_{x_i=a_i}}(e_u - e_v)$. A similar argument shows that $\phi(y)$ also has such a decomposition. We now argue the properties of ϕ.

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 (SI_d, $\mathcal{P}_d$)**.** Let $\mathbb{F}$ be a finite field. Denote by SI_d, the Boolean function defined on $\mathbb{F}^{d\times d} \times \mathbb{F}^{d\times d} \to \{0, 1\}$ as for any $A, B \in \mathbb{F}^{d\times d}$ $\mathsf{SI}_d(A, B) = 1$ if and only if $\mathsf{rowspan}(A) \cap \mathsf{rowspan}(B) \neq \{0\}$. Note that the row span 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 $(x, y) \in \{0, 1\}^n \times \{0, 1\}^n$, $\mathsf{IP}_n(x, y) = \sum_{i=1}^n x_i y_i \mod 2$, $\mathsf{EQ}_n(x, y)$ is 1 if $\forall i \in [n]\ x_i = y_i$ and is 0 otherwise, $\mathsf{INEQ}_n(x, y) = \neg\mathsf{EQ}_n(x, y)$ and $\mathsf{DISJ}_n(x, y) = 1$ 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 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_is and y_is 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 $\mathsf{PAL}_n^q(x, y) = 1$ 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 $d \in \mathbb{N}$, $\mathcal{P}_d$ appears as an induced subgraph of the bipartite realization of SI_d. Hence, $\mathsf{pd}(\mathsf{SI}_d) \ge \mathsf{pd}(\mathcal{P}_d)$.

We need the following definition of Gaussian coefficients. For non-negative integers n, k and a prime power q, $\begin{bmatrix} n \\ k \end{bmatrix}_q$ is the expression, $\frac{(q^n-1)(q^n-q)\ldots(q^n-q^{k-1})}{(q^k-1)(q^k-q)\ldots(q^k-q^{k-1})}$ if $n \ge k, k \ge 1$, 0 if $n < k, k \ge 1$, 1 if $n \ge 0, k = 0$.

Linear Algebra : We recall some basic lemmas from linear algebra which we use later. Unless otherwise mentioned, all our algebraic formulations are over finite fields ($\mathbb{F}$ of size q). For vector spaces V_1, V_2 with dimensions k_1, k_2 respectively, the *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 Span $\left(\{uv^\top \mid u \in U, v \in V\}\right)$ 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. 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 $B = \{u_i v_j^\top \mid i \in \{1, 2, \ldots, k\}, j \in \{1, 2, \ldots, \ell\}\}$ 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, $(U_1 \otimes U_2) \cap (V_1 \otimes V_2) \neq \{0\} \iff U_1 \cap V_1 \neq \{0\}$ and $U_2 \cap V_2 \neq \{0\}$*

The proofs of the two Propositions 2.4 and 2.5 are fairly elementary and follows from basic linear algebra. For example Proposition 2.5 follows as an easy corollary from an exercise from [19, Chapter 14, exercise 12][6].

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 Π_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 $A, B \subseteq_S \mathbb{F}^d$, 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. We start by proving closure properties of projective dimension under Boolean operations $\wedge$ and $\vee$. The proof is based on direct sum and tensor product of vector spaces.

Lemma 3.1. *Let $\mathbb{F}$ be an arbitrary field. For any two functions $f_1 : \{0,1\}^{2n} \to \{0,1\}$, $f_2 : \{0,1\}^{2n} \to \{0,1\}$, $\mathsf{pd}_{\mathbb{F}}(f_1 \vee f_2) \leq \mathsf{pd}_{\mathbb{F}}(f_1) + \mathsf{pd}_{\mathbb{F}}(f_2)$ and $\mathsf{pd}_{\mathbb{F}}(f_1 \wedge f_2) \leq \mathsf{pd}_{\mathbb{F}}(f_1) \cdot \mathsf{pd}_{\mathbb{F}}(f_2)$*

Proof. In this proof, for a Boolean f with bipartite representation $G_f(U, V, E)$ we define the map ϕ 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 $\phi_1 : \{0,1\}^n \times \{0,1\} \to \mathbb{F}^{k_1}, \phi_2 : \{0,1\}^n \times \{0,1\} \to \mathbb{F}^{k_2}$ respectively.

- From ϕ_1 and ϕ_2 we construct a subspace assignment $\phi : \{0,1\}^n \times \{0,1\} \to \mathbb{F}^{k_1+k_2}$ which realize $f_1 \vee f_2$ 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 $v \in$

[6] There is a typo in the way the exercise is stated in [19, Chapter 14, exercise 12]. For this reason we give a proof of this result in Appendix C.

$\{0,1\}^n, \phi(v,1) = \phi_1(v,1) \oplus \phi_2(v,1)$. 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 ϕ_1 and ϕ_2 we construct a subspace assignment $\phi : \{0,1\}^n \times \{0,1\} \to \mathbb{F}^{k_1 k_2}$, realizing $f_1 \wedge f_2$ thus proving the theorem. Consider the following projective dimension assignment ϕ: 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.

□

The $\vee$ part of the above lemma was also observed (without proof) in [17]. A natural question is whether we can improve any of the above bounds. In that context, we make the following remarks: (1) the construction for $\vee$ is tight up to constant factors, (2) we cannot expect a general relation connecting $\mathsf{pd}_{\mathbb{R}}(f)$ and $\mathsf{pd}_{\mathbb{R}}(\neg f)$.

- We prove that the construction for $\vee$ is tight up to constant factors. Assume that n is a multiple of 4. Consider the functions $f(x_1, \ldots, x_{\frac{n}{4}}, x_{\frac{n}{2}+1}, \ldots, x_{\frac{3n}{4}})$ and $g(x_{\frac{n}{4}+1}, \ldots, x_{\frac{n}{2}}, x_{\frac{3n}{4}+1}, \ldots, x_n)$ 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 $f \vee g$ is the inequality function on $\frac{n}{2}$ variables $x_1, \ldots, x_{\frac{n}{2}}$ and the next $\frac{n}{2}$ 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 $\mathsf{pd}(f) \leq n$ and $\mathsf{pd}(g) \leq n$. Hence by Lemma 3.1, $\mathsf{pd}(f \vee g) \leq \mathsf{pd}(f) + \mathsf{pd}(g) \leq 2n$. Lower bound on projective dimension of inequality function comes from [16, Theorem 4], giving $\mathsf{pd}(f \vee g) \geq \epsilon . \frac{n}{2}$ for an absolute constant ϵ. This shows that $\mathsf{pd}(f \vee g) = \Theta(n)$.
- A natural idea to improve the upper bound of $\mathsf{pd}(f_1 \wedge f_2)$ is to prove upper bounds for $\mathsf{pd}(\neg f)$ in terms of $\mathsf{pd}(f)$. However, we remark that over $\mathbb{R}$, it is known [16] that $\mathsf{pd}_{\mathbb{R}}(\mathsf{INEQ}_n)$ is $\Omega(n)$ while $\mathsf{pd}_{\mathbb{R}}(\mathsf{EQ}_n) = 2$. Hence we cannot expect a general relation connecting $\mathsf{pd}_{\mathbb{R}}(f)$ and $\mathsf{pd}_{\mathbb{R}}(\neg f)$.

We now observe a characterization of bipartite graphs having projective dimension at most d over $\mathbb{F}$. Let $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, and $G_f(X, Y, E)$ be its bipartite realization. Let $\mathsf{pd}(G_f) = d$.

Proposition 3.2. *For any subspace assignment ϕ 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 ϕ is no longer a realization of G_f since z should not have been adjacent to x'. □

Lemma 3.3 (Characterization). *Let G be a bipartite graph with no two vertices having same neighborhood, $\mathsf{pd}(G) \leq d$ 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, $\mathsf{pd}(G) \leq d$, 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 $\mathsf{pd}(G) \leq d$. 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 3.2, 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. □

It follows that $\mathsf{pd}(\mathcal{P}_d) \leq d$. Observe that, in any projective assignment, the vertices with different neighborhoods should be assigned different subspaces. For $\mathsf{pd}(\mathcal{P}_d)$, 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 3.4. *For any $d \in \mathbb{N}$, $\mathsf{pd}(\mathcal{P}_d) = \mathsf{pd}(\mathsf{SI}_d) = d$.*

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 $\mathsf{pd}(\mathcal{P}_d)$ cannot be used to obtain more than a $\sqrt{\log N}$ lower bound for $\mathsf{pd}(G)$. Also, for many functions, the number of vertices of distinct neighborhood can be smaller.

We observe 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 a matching connecting vertices in A_1 to B_2 and A_2 to B_1 respectively. Since the matching edges associated with every vertex is unique, the neighborhoods of all vertices are bound to be distinct. Applying Lemma 3.1 and observing that matching (i.e, equality function) has projective dimension at most n, $\mathsf{pd}(f') \leq 2\mathsf{pd}(f) + 2n$. 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 $\mathsf{upd}(f)$) d over field $\mathbb{F}$, if d is the smallest possible dimension for which there exists a vector space K of dimension d over $\mathbb{F}$ with a map ϕ 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 $(u, v) \in U \times V$, $\mathsf{dim}\,(\phi(u) \cap \phi(v)) \leq 1$.

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, $\mathsf{upd}_{\mathbb{F}}(f) \leq s$ for any field $\mathbb{F}$.*

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

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

Lemma 4.3. *Let M be the bipartite adjacency matrix of $\mathcal{P}_d$, then $\mathsf{rank}\,(M) \geq \begin{bmatrix} d \\ d/2 \end{bmatrix}_q \geq q^{\frac{d^2}{4}}$*

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 $\overline{W_{ik}}$ over $\mathbb{R}$ as $\overline{W_{ik}}(I, K) = 1$ if $I \cap K = \{0\}$ and 0 otherwise. This matrix has dimension $\begin{bmatrix} d \\ i \end{bmatrix}_q \times \begin{bmatrix} d \\ k \end{bmatrix}_q$.

Consider the submatrix M_i of M with rows and columns indexed by subspaces of dimension exactly i. Observe that $\overline{W_{ii}} = J - M_i$ where J is an all ones matrix of appropriate order. These matrices are well-studied (see [8]). Closed form expressions for eigenvalues are computed in [6, 13] and the eigenvalues are known to be non-zero. Hence for $0 \leq i \leq d/2$ the matrix $\overline{W_{ii}}$ has rank $\begin{bmatrix} d \\ i \end{bmatrix}_q$. Since $\overline{W_{ii}} = J - M_i$, $\mathsf{rank}\,(M_i) \geq \mathsf{rank}\left(\overline{W_{ii}}\right) - 1$. This shows that $\mathsf{rank}\,(M) \geq \mathsf{rank}\,(M_i) = \begin{bmatrix} d \\ i \end{bmatrix}_q$ for all i such that $2i \leq d$. Choosing $i = d/2$ gives $\mathsf{rank}\,(M) \geq \begin{bmatrix} d \\ d/2 \end{bmatrix}_q - 1 \geq q^{\frac{d^2}{4}} - 1$. □

We now present two approaches for showing lower bounds on $\mathsf{upd}(f)$ - one using intersection families of vector spaces and the other using rectangle arguments on M_f.

Lower Bound for $\mathsf{upd}(\mathcal{P}_d)$ using intersecting families of vector spaces : To prove a lower bound on $\mathsf{upd}(\mathcal{P}_d)$ 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 $D = \mathsf{upd}(\mathcal{P}_d)$. We first show that $\mathsf{rank}\,(N)$ is at most $1 + \begin{bmatrix} D \\ 1 \end{bmatrix}_q$. Then by Lemma 4.3 we get that $\mathsf{rank}\,(N)$ 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 $N[G_r, H_t] = p(\mathsf{dim}(G_r \cap H_t))$ with $G_r \in \mathcal{G}$, $H_t \in \mathcal{H}$, then $\mathsf{rank}\,(N) \leq \sum_{i=0}^{s} \begin{bmatrix} D \\ i \end{bmatrix}_q$*

Proof. This proof is inspired by the proof in [7] of a similar claim where a non-bipartite version of this lemma is proved. To begin with, note that p is a degree s polynomial in q^x, and hence can be written as a linear combination of polynomials $p_i = \begin{bmatrix} x \\ i \end{bmatrix}_q, 0 \leq i \leq s$. 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 $N_i[G_r, H_s] = p_i(\mathsf{dim} G_r \cap H_s)$. 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 Γ_j corresponding to $\mathcal{G}$ and Δ_j corresponding to $\mathcal{H}$ defined as $\Gamma_j(G, I) = 1$ if $\mathsf{dim}(I) = j, G \in \mathcal{G}, I \subseteq_s G$ and 0 otherwise. $\Delta_j(H, I) = 1$ if $\mathsf{dim}(I) = j, H \in \mathcal{H}, I \subseteq_s H$ and 0 otherwise. Note that rank of the these matrices are upper bounded by the number of columns which is $\begin{bmatrix} D \\ j \end{bmatrix}_q$. We claim that for any $i \in \{0, 1, \ldots, s\}$, $\mathsf{rank}\,(N_i) \leq \begin{bmatrix} D \\ i \end{bmatrix}_q$. 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

$N_i = \Gamma_i \Delta_i^T$. Hence $\mathsf{rank}\,(N_i) \leq \min\{\mathsf{rank}\,(\Gamma_i)\,, \mathsf{rank}\,(\Delta_i)\} \leq \begin{bmatrix} D \\ i \end{bmatrix}_q$. For $(G_r, H_t) \in \mathcal{G} \times \mathcal{H}$,

$$\begin{aligned}
\Gamma_i \Delta_i^T(G_r, H_t) &= \sum_{I \in \mathcal{F}_i} \Gamma_i(G_r, I)\Delta_i^T(I, H_t) \\
&= \sum_{I \in \mathcal{F}_i} \Gamma_i(G_r, I)\Delta_i(H_t, I) \\
&= \sum_{I \in \mathcal{F}_i} [I \subseteq_s G_r] \wedge [I \subseteq_s H_t] \\
&= \sum_{I \in \mathcal{F}_i} [I \subseteq_s G_r \cap H_t] \\
&= \begin{bmatrix} \mathsf{dim}(G_r \cap H_t) \\ i \end{bmatrix}_q = N_i(G_r, H_t)
\end{aligned}$$

□

We apply Lemma 4.4 on N defined via $p(x) = q^x - 1$ with $s = 1$, to get $q^{d^2/4} \leq \begin{bmatrix} d \\ d/2 \end{bmatrix}_q \leq 1 + \begin{bmatrix} D \\ 1 \end{bmatrix}_q = 1 + (q^D - 1)/(q - 1)$. By definition, $\mathsf{rank}\,(N) = \mathsf{rank}\,(M)$. This gives that $D = \Omega(d^2)$ and proves Theorem 1.1.

Lower Bound for $\mathsf{upd}(\mathcal{P}_d)$ from Rectangle Arguments : We now give an alternate proof of for Theorem 1.1 using combinatorial rectangle arguments.

Lemma 4.5. *For $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ with M_f denoting the bipartite adjacency matrix of G_f, $rank_{\mathbb{R}}(M_f) \leq q^{O(\mathsf{upd}_{\mathbb{F}}(f))}$ where $\mathbb{F}$ is a finite field of size q.*

Proof. Let ϕ be a subspace assignment realizing f of dimension d with intersection dimension 1. Let $S(v)$ for $v \in \mathbb{F}_q^d$ denote $\{(a,b) \in \{0,1\}^n \times \{0,1\}^n \mid \phi(a) \cap \phi(b) = \mathsf{span}\,\{v\}\}$. 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, $rank(M_f) \leq \sum_{v \in T} rank(M_v)$. Since $rank(M_v) = 1$, $rank(M_f) \leq |T|$. The fact that the number of 1 dimensional spaces in $\mathbb{F}^d$ can be at most $\frac{q^d - 1}{q - 1}$ completes the proof. Note that the rank of M_f can be over any field (we choose $\mathbb{R}$). □

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 $\mathsf{upd}(f) = \Omega(n)$. There are several Boolean functions with this property, well-studied in the context of communication complexity (see textbook [11]). Hence, we have for $f \in \{\mathsf{IP}_n, \mathsf{EQ}_n, \mathsf{INEQ}_n, \mathsf{DISJ}_n, \mathsf{PAL}_n^q\}$, $\mathsf{upd}_{\mathbb{F}}(f)$ is $\Omega(n)$ for any finite field $\mathbb{F}$.

For arguing about PAL_n^q, it can be observed that the graph is strongly regular (as $q \equiv 1 \mod 4$) and hence the adjacency matrix has full rank over $\mathbb{R}$ [5]. Except for PAL_n^q, all the above functions have $O(n)$ sized deterministic branching programs computing them and hence the Pudlák-Rödl theorem (Theorem 2.1) gives that upd for these functions (except PAL_n^q) are $O(n)$ and hence the above lower bound is indeed tight.

From Lemma 4.3, it follows that the function SI_d also has rank $2^{\Omega(d^2)}$. To see this, it suffices to observe that $\mathcal{P}_d$ appears as an induced subgraph in the bipartite realization of SI_d. Thus, $\mathsf{upd}(\mathsf{SI}_d)$

is $\Omega(d^2)$. We proved in Theorem 3.4 that $\mathsf{pd}(\mathsf{SI}_d) = d$. This establishes a quadratic gap between the two parameters. This completes the proof of Theorem 1.1.

Let $\mathsf{D}(f)$ 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 4.6. If $\mathsf{upd}(f) \leq D$, the assignment also gives a partitioning of the 1s in M_f into at most $\frac{q^D-1}{q-1}$ 1-rectangles. However, it is unclear whether this immediately gives a similar partition of 0s into 0-rectangles as well. Notice that if $\mathsf{D}(f) \leq d$, 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 4.6. *For a Boolean function* $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ *and a finite field* $\mathbb{F}$, $\mathsf{upd}_{\mathbb{F}}(f) \leq 2^{\mathsf{D}(f)}$ *and* $\mathsf{D}(f) \leq (\mathsf{pd}_{\mathbb{F}}(f))^2 \log |\mathbb{F}|$

Proof. We give a proof of the first inequality. Any deterministic communication protocol computing f of cost $\mathsf{D}(f)$, partitions M_f into k rectangles where $k \leq 2^{\mathsf{D}(f)}$ rectangles. Define $f_i : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ for each rectangle R_i $i \in [k]$, such that $f_i(x,y) = 1$ iff $(x,y) \in R_i$. Note that $\mathsf{upd}_{\mathbb{F}}(f_i) = 1$ and $f = \vee_{i=1}^k f_i$. 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 $f_1, \vee f_2 \vee \ldots f_k$ will have the property that for any (x,y) with $f(x,y) = 1$, the intersection dimension is 1. Hence $\mathsf{upd}_{\mathbb{F}}(f) \leq k \leq 2^{\mathsf{D}(f)}$. To prove the second inequality, consider the protocol where Alice sends the subspace associated with her input as a $\mathsf{pd}_{\mathbb{F}}(f) \times \mathsf{pd}_{\mathbb{F}}(f)$ matrix. Bob then checks if this subspace intersects with his own subspace and sends 1 if it does so and sends 0 otherwise. □

An immediate consequence of Proposition 4.6 is that all symmetric functions f on $2n$ bits have have projective dimension $O(n)$. Note that the first inequality is tight, up to constant factors in the exponent. To see this, consider the function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ whose $\mathsf{pd}_{\mathbb{F}}(f) = \Omega(2^{n/2})$ [16, Proposition 1] and note that $\mathsf{D}(f)$ for any f is at most n. Tightness of second inequality is witnessed by SI_d since by Lemma 4.3, $\mathsf{D}(\mathsf{SI}_d) = \Omega(d^2)$ while $\mathsf{pd}(\mathsf{SI}_d) = d$.

5 Bitwise Decomposable Projective Dimension

The restriction of intersection dimension being 1, although 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. 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.

Proposition 5.1. *(Implicit in Remark 1.30 of [9]) There exist a function* $f : \{0,1\}^n \times \{0,1\}^n$ *that requires size* $\Omega(\frac{2^n}{n})$ *for any branching program computing* f *but the* $\mathsf{upd}(f) \leq O(n)$ *for any balanced partitioning of the input into two parts.*

Proof. Consider the function EQ_n. The graph $G_{\mathsf{EQ}_n}(U,V,E)$ 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 $\mathcal{G}$ of $N!$ different labeled graphs. Let $\mathcal{F}$ be the set of Boolean functions whose corresponding graph is in $\mathcal{G}$ (or equivalently $\mathcal{F}$ of $N!$ different functions). Let t be the smallest number such that any function in $\mathcal{F}$ can be computed by a branching program of size at most t. The number of branching programs of size $\leq t$ (bounded by $O(t^t)$ [9]) forms an upper bound on $|\mathcal{F}|$. 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 $f \in \mathcal{F}$ such that $\mathsf{upd}(f) = \mathsf{upd}(\mathsf{EQ}_n) \leq n$ but $\mathsf{bpsize}(f)$ is $\Omega\left(\frac{2^n}{n}\right)$ for this partition.

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

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

The above proposition can be shown by adapting the counting argument presented in Remark 1.30 of [9].

5.1 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 *bit projective dimension*, $\mathsf{bitpdim}(G) \leq d$, 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 a projective assignment ϕ 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 $(x,y) \in \{0,1\}^n \times \{0,1\}^n$, $\phi(x) = \underset{1 \leq i \leq n}{\mathsf{span}} \{U_i^{x_i}\}$, $\phi(y) = \underset{1 \leq j \leq n}{\mathsf{span}} \{V_j^{y_j}\}$ and f is realized by ϕ.
2. Let $S = \{e_i - e_j \mid i,j \in [d], i \neq j\}$. For any $W \in \mathcal{C} \cup \mathcal{D}$, $\exists S' \subseteq S$ such that $W = \mathsf{span}\{S'\}$.
3. for any $S_1, S_2 \subseteq ([n] \times \{0,1\})$ such that $S_1 \cap S_2 = \phi$, $\underset{(i,a) \in S_1}{\mathsf{span}} \{U_i^a\} \cap \underset{(j,b) \in S_2}{\mathsf{span}} \{U_j^b\} = \{0\}$. Same property must hold for subspaces in $\mathcal{D}$.

We show that the new parameter bitwise decomposable projective dimension ($\mathsf{bitpdim}$) tightly characterizes the branching program size, up to constants in the exponent.

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

Proof. The subspace assignment obtained by applying (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 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 $y_1 = 0$ and another which checks $y_1 = 1$ (see Figure 1).

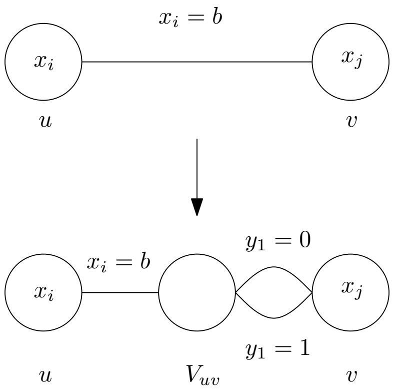


Figure 1: Edge modification

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 $|E(B')| \leq 2(|V(B')|)$. 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 ϕ be projective assignment obtained from B'' via Pudlák-Rödl theorem. We claim that ϕ satisfies all the requirements of bitpdim(f).

1. Since ϕ 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 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 $\mathsf{bitpdim}(f) \leq d$, 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 $\mathcal{C}, \mathcal{D}$ of the $\mathsf{bitpdim}$ assignment as the advice string. These can be specified by a list of n basis matrices each of size d^2. Note that for any any f which has a polynomial sized branching program, $d = \mathsf{bitpdim}(f)$ is at most $poly(n)$, and hence the advice string is $poly(n)$ sized and depends only on n.

We construct a deterministic branching program computing f as follows. On input x, y, from the basis matrices in $\mathcal{C}, \mathcal{D}$, construct an undirected graph[7] G^* with all standard basis vectors in $\mathcal{C}, \mathcal{D}$ as vertices and add an edge between two vertices u, v if $e_u - e_v \in U_i^{x_i}$ or $e_u - e_v \in V_j^{y_j}$ for $i, j \in [n]$. For input x, y, $f(x, y) = 1$ iff G^* has a cycle. To see this, let $C = C_1 \cup C_2$ be a cycle in G^* where C_1 consists of edges from basis matrices in $\mathcal{C}$ and C_2 contain edges from basis matrices in $\mathcal{D}$. Note that if one of C_1 or C_2 is empty then there is a cycle consisting only of vectors from $\mathcal{C}$ which implies a linear dependence among vectors in $\mathcal{C}$. But this contradicts Property 3 of $\mathsf{bitpdim}$ assignment. Hence both C_1 and C_2 are non-empty.

Then, it must be that $\sum_{(u,v)\in C_1} e_u - e_v + \sum_{(w,z)\in C_2} e_w - e_z = 0$. Hence $\sum_{(u,v)\in C_1} e_u - e_v = -\sum_{(w,z)\in C_2} e_w - e_z$. 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 [18]. The log-space algorithm can also be converted to an equivalent branching program of size n^c for a constant c.

We can improve the constant c to $3 + \epsilon$. We achieve this using the well known random walk based RL algorithm for reachability [1], amplifying the error and suitably fixing the random bits to achieve a non-uniform branching program of size $d^{3+\epsilon}$.

The RL algorithm requires to store $\log d$ bits to remember the current vertex while doing the random walk and another $\log d$ bits to store the next vertex in the walk. It performs a walk of length $4d^3$ and answers correctly with probability of $1/2$ [14]. Amplifying the error does not incur any extra space as the algorithm has a one-sided error and it never errs when it accepts. This gives a probabilistic Turing machine using $2 \log d + 1$ work space. By amplifying the success probability, we can obtain a choice of a random bits which works for all inputs of a fixed length. The conversion of this machine to a branching program will incur storing of the head index position of the work tape and input tape position which incur an additional $\log \log d + \log d$ space. Hence overall space is $3 \log d + \log \log d = (3 + \epsilon) \log d$ for small fixed $\epsilon > 0$, thus proving that $c \leq 3 + \epsilon$.

□

Assuming $\mathsf{C_{=}L} \not\subseteq \mathsf{L/poly}$, the function SI_d (a language which is hard for $\mathsf{C_{=}L}$ under Turing reductions) cannot be computed by deterministic branching programs of polynomial size.

Proposition 5.5. *The function family $\{\mathsf{SI}_d\}_{d\geq 0}$ is hard for $\mathsf{C_{=}L}$ via logspace Turing reductions. Moreover, the negation of $\{\mathsf{SI}_d\}_{d\geq 0}$ is in $\mathsf{L}^{\mathsf{C_{=}L}}$ (and hence in NC^2).*

Proof. We start with the following claim.

[7] Note that this is not a deterministic branching program.

Claim 5.6 (Corollary 2.3 of [2]). *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 $\mathbb{F}^{n\times n}$, $det(M) = 0 \implies rank(g(M)) = n$ and $det(M) \neq 0 \implies rank(g(M)) = n - 1$*

Consider the language $L = \{(M_1, M_2) \mid \mathsf{rowspan}(M_1) \cap \mathsf{rowspan}(M_2) \neq \{0\}, M_1, M_2 \in \mathbb{F}^{d\times d}\}$. The reduction is as follows. Given an $M \in \mathbb{F}^{d\times d}$, apply g (defined in Claim 5.6) on M to get N, and define for $1 \leq i \leq d$, $H^i = (M_1^i, M_2^i)$ where M_1^i is the matrix consisting of i^{th} row of N repeated n times and M_2^i 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 $det(M)$ is 0, then $N = g(M)$ (by Claim 5.6) must have full rank. Hence for all $1 \leq i \leq d$, $\mathsf{rowspan}(M_1^i)$ and $\mathsf{rowspan}(M_2^i)$ does not intersect. If $det(M) \neq 0$, then $N = g(M)$ (by Claim 5.6) must have a linearly dependent column and hence there is some i for which $\mathsf{rowspan}(M_1^i)$ and $\mathsf{rowspan}(M_2^i)$ is non-zero. Also the overall reduction runs in logspace as g is logspace computable.

The upper bound follows by observing that given two $d \times d$ matrices M_1 and M_2, their individual ranks r_1 and r_2 can be computed in $\mathsf{L}^{\mathsf{C}_=\mathsf{L}}$ [2]. Consider the matrix M of size $d \times 2d$ by adjoining M_1 and M_2. It follows that the $rowspace(M_1) \cap rowspace(M_2) \neq \phi$ if and only if $\mathsf{rank}\,(M) < r_1 + r_2$. The latter condition can also be tested using a query to $\mathsf{C}_=\mathsf{L}$ oracle. □

5.2 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. As mentioned in the introduction, 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 [15] 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 $\mathsf{bitpdim}$ assignment ϕ, consider the restriction of f denoted f_ρ where ρ fixes all variables except the ones in T_i to 0 or 1 where T_i is some subset of variables in the left partition. For different restrictions ρ, we are guaranteed to get at least $c_i(f)$ different functions. We show that for each restriction ρ, we can obtain an assignment from ϕ realizing f_ρ. Hence the number of different $\mathsf{bitpdim}$ assignments for ρ 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 $\mathsf{bitpdim}$ 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 subspaces associated with T_is are disjoint, we get a lower bound on d as $d = \sum_i d_i$.

Theorem 5.7. *For a Boolean function $f : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ on $2n$ variables, let $T_1, \ldots, T_m$ are partition of variables to m blocks of size r_i on the first n variables. Let $c_i(f)$ be the number of distinct sub functions of f when restricted to T_i, then* $\mathsf{bitpdim}(f) \geq \sum_{i=1}^{m} \frac{\log c_i(f)}{\log(\log c_i(f))}$

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

Define $L = \underset{i\in[n],\rho(i)\neq *}{\mathsf{span}}\{U_i^{\rho(i)}\}$ and $R = \underset{j\in[n]}{\mathsf{span}}\{V_j^{\rho(n+j)}\}$. For any $x \in \{0,1\}^n$ that agrees with ρ on the first n bits, define $Z^x = \underset{j\in T_i}{\mathsf{span}}\{U_j^{x_j}\}$ Note that for any (x,y), which agrees with ρ, has $\phi(x) = L + Z^x$ and $\phi(y) = R$. For any $f_{\rho_1} \not\equiv f_{\rho_2}$, $G_{f_{\rho_1}} \neq G_{f_{\rho_2}}$. Hence the number of $\mathsf{bitpdim}$ assignments is at least the number of different sub functions. We need to give a $\mathsf{bitpdim}$ assignment for $G_{f_\rho}(V_1, V_2, E)$ where $V_1 = \{x \mid x \text{ agrees with } \rho\}$, $V_2 = \{y\}$ where $y = \rho_{[n+1,\ldots,2n]}$ and $E = \{(x,y) | x \in V_1, y \in V_2, f(x,y) = 1\}$. We use the following property to come up with such an assignment.

Property 5.8. *Let ρ 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 ρ and $\forall x, y \in \{0,1\}^n$ which agrees with ρ, 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 $v \neq \vec{0}$.*

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 $v = \vec{0}$. Since $\vec{0} \in Z^{\hat{x}}$ for any $\hat{x}$, $w = u + \vec{0}$ is in $L + Z^{\hat{x}}$ and R. Thus the function after restriction ρ is a constant. This contradicts the choice of ρ. □

The assignment ψ_ρ for G_{f_ρ} defined as : $\psi_\rho(x) = Z^x$ and $\psi_\rho(y) = \underset{x\in V_1}{\mathsf{span}}\{\Pi_{Z^x}(R \cap (L + Z^x))\}$ Note that for $(x,y) \in V_1 \times V_2$, $f_\rho(x) = f(x,y)$. Following claim shows that ψ_ρ realize f_ρ.

Claim 5.9. *For any $(x,y) \in V_1 \times V_2$, $f(x,y) = 1$ if and only if $\psi_\rho(x) \cap \psi_\rho(y) \neq \{0\}$.*

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 suffices to prove : $\forall (x,y) \in V_1 \times V_2$, $\psi_\rho(x) \cap \psi_\rho(y) \neq \{0\} \iff \phi(x) \cap \phi(y) \neq \{0\}$.

We first prove that $\psi_\rho(x) \cap \psi_\rho(y) \neq \{0\}$ implies $\phi(x) \cap \phi(y) \neq \{0\}$. Let v be a non-zero vector in $\psi_\rho(x) \cap \psi_\rho(y)$. By definition of $\psi_\rho(x)$, $v \in Z^x$. By definition of $\psi_\rho(y)$, 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 $u_{\hat{x}}$ 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 $\psi_\rho(x) \cap \psi_\rho(y) \neq \{0\}$. 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 5.8 we have $v \neq \vec{0}$. By definition $v \in \psi_\rho(y)$. Along with $v \in Z^x$, we get $\psi_\rho(x) \cap \psi_\rho(y) \neq \{0\}$. □

Let $Z = \underset{j\in T_i}{\mathsf{span}}\{U_j^0 + U_j^1\}$. We now prove that subspace assignment on the only vertex in the right partition of G_ρ which is $\underset{x\in V_1}{\mathsf{span}}\{\Pi_{Z^x}(R)\}$ is indeed $\Pi_Z(R)$.

Claim 5.10. $\Pi_Z(R) = \underset{x\in V_1}{\mathsf{span}}\{\Pi_{Z^x}(R)\}$

Proof. We show $\underset{x\in V_1}{\mathsf{span}}\{\Pi_{Z^x}(R)\} \subseteq \Pi_Z(R)$. Note that $\underset{x\in V_1}{\mathsf{span}}\{\Pi_{Z^x}(R)\} = \underset{x\in V_1, w\in R}{\mathsf{span}}\{\Pi_{Z^x}(w)\}$. 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 $\{U_i^b\}_{i\in[n],b\in\{0,1\}}$

are disjoint, $\Pi_{Z^x}(w) = +_{i\in[n],\rho(i)=*}\Pi_{U_i^{x_i}}(w)$. As $Z = \underset{j\in T_i}{\mathsf{span}}\{U_j^0 + U_j^1\}$, every $\Pi_{U_i^{x_i}}(w) \in \Pi_Z(R)$. Hence the span is also in $\Pi_Z(R)$.

Now we show that $\Pi_Z(R) \subseteq \underset{x\in V_1}{\mathsf{span}}\{\Pi_{Z^x}(R)\}$. Let $T_i = \{i_1, \ldots, i_k\}$. For $1 \leq j \leq k$ define x^j to be $x+e^j$ where $x \in \{0,1\}^n$ agrees with ρ and for any index $i \in [n]$ with $\rho(i) = *$, $x_i = 0$ and $e^j \in \{0,1\}^n$ is 0 at every index other than i_j. Note that for any $j_1 \neq j_2, j_1, j_2 \in T_i$, $Z^{x^{j_1}} \cap Z^{x^{j_2}} = \{0\}$ by Property 3 of Definition 5.2) Also note that $\underset{j\in T_i}{\mathsf{span}}\{Z^{x^j}\} = \underset{j\in T_i}{\mathsf{span}}\{U_j^{x_j}\} = Z$. Hence, $\Pi_Z(R) = \underset{j\in T_i}{\mathsf{span}}\{\Pi_{Z^{x^j}}(R)\}$. But $\underset{j\in T_i}{\mathsf{span}}\{\Pi_{Z^{x^j}}(R)\} \subseteq \underset{x\in V_1}{\mathsf{span}}\{\Pi_{Z^x}(R)\}$. □

For any ρ, which fixes all variables outside T_i, Z is the same. And since there is only one vertex on the right partition, for different ρ, ρ', $\Pi_Z(R_\rho) = \Pi_Z(R_{\rho'})$ implies $\psi_\rho = \psi_{\rho'}$. Hence to count the number of different ψ_ρ's for different f_ρ'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 5.11. *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 $\Pi_Z(R) = \mathsf{span}\{S'\}$.*

Proof. By the property of the $\mathsf{bitpdim}$ assignment, $\forall i \in [n]$ and $\forall b \in \{0,1\}$, $V_i^b = \mathsf{span}\left\{F_i^b\right\}$ where F_i^b is a collection of difference of standard basis vectors. Recall that $R = \underset{j\in[n]}{\mathsf{span}}\{V_j^{\rho(n+j)}\}$. Let $F = \left\{(e_u - e_v) \mid e_u - e_v \in F_j^{\rho(n+j)}, j \in [n]\right\}$. Since projections are linear maps and the fact that $F_j^{\rho(n+j)}$ spans $V_j^{\rho(n+j)}$ we get that, $\Pi_Z(R) = \underset{w\in F}{\mathsf{span}}\{\Pi_Z(w)\}$. Since Z is also a span of difference of standard basis vectors, $\Pi_Z(e_u - e_v)$ is one of $\vec{0}$, $e_u - e_w$ 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, $\underset{e_u - e_v\in S''}{\mathsf{span}}\{e_u - e_v\} = \Pi_Z(R)$. Choose S' as a linear independent subset of S''. □

Property 5.11 along with the fact that $\Pi_Z(R)$ is a subspace of Z, gives us that the number of different $\Pi_Z(R)$ is upper bounded by number of different subsets S' of S such that $|S'| = d_i$ where $d_i = \mathsf{dim}(Z)$. As S' is a set of difference of standard basis vectors from Z, $|S'| \leq \binom{d_i}{2}$. Thus the number of different such S' are at most $\sum_{k=0}^{d_i}\binom{d_i^2}{k} = 2^{O(d_i \log d_i)}$.

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

Theorem 5.7 gives a super linear lower bound for Element Distinctness function. From a manuscript by Beame et.al, ([4], see also [9], Chapter 1), we have $c_i(ED_n) \geq 2^{n/2}/n$. Hence applying this count to Theorem 5.7, we get that $d \geq \Omega\left(\frac{n}{\log n} \cdot \frac{n}{\log n}\right) = \Omega\left(\frac{n^2}{(\log n)^2}\right)$.

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 SI_d.

Lemma 5.12. *For any $i \in [d]$, there are $2^{\Omega(d^2)}$ different restrictions ρ of SI_d which fixes all entries other than ith row of the $d \times d$ matrix in the left partition.*

Proof. Fix any $i \in [d]$. Let S be a subspace of $\mathbb{F}_2^d$. Define ρ_S to be $\mathsf{SI}_d(\mathbf{A}, B)$ where B is a matrix whose rowspace is S. And $\mathbf{A}$ 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 $\mathbf{A}(x)$ is $\mathsf{span}\,\{v\}$.

We claim that for any $S, S' \subseteq_S \mathbb{F}_2^d$ where $S \neq S'$, $(\mathsf{SI}_d)_{\rho_S} \not\equiv (\mathsf{SI}_d)_{\rho'_S}$. By definition $(\mathsf{SI}_d)_{\rho_S} \equiv \mathsf{SI}_d(\mathbf{A}, B)$ and $(\mathsf{SI}_d)_{\rho'_S} \equiv \mathsf{SI}_d(\mathbf{A}, B')$ 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 $\mathsf{SI}_d(\mathbf{A}(v), B) = 1$ as $v \in S$ where as $\mathsf{SI}_d(\mathbf{A}(v), B') = 0$ 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 SI_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 SI_d.

6 Standard Variants of Projective Dimension

In this section, 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.

Definition 6.1 (**Standard Projective Dimension**)**.** 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 standard projective dimension (denoted by $\mathsf{spd}(f)$) d over field $\mathbb{F}$, if d is the smallest possible dimension for which there exists a vector space K of dimension d over $\mathbb{F}$ with a map ϕ 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 $(u, v) \in U \times V$, $\mathsf{dim}\,(\phi(u) \cap \phi(v)) \leq 1$, we say that the *standard projective dimension is with intersection dimension* 1, denoted by $\mathsf{uspd}(f)$. We make some easy observations about the definition itself.

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, $\mathsf{spd}(G) \leq \mathsf{uspd}(G) \leq m$.

Even though $\mathsf{pd}(G) \leq \mathsf{spd}(G)$, there are graphs for which the gap is exponential. For example, consider the bipartite realization G of EQ_n with $N = 2^n$. We know $\mathsf{pd}(G) = \Theta(\log N)$ but $\mathsf{spd}(G) \geq N$ 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.

Definition 6.2 (Biclique cover number)**.** For a graph G, a collection of complete bipartite graphs defined on $V(G)$ is said to cover G if every edge in G is present in some complete bipartite graph of the collection. The size of the smallest collection of bipartite graph which covers G is its biclique cover number (denoted by $\mathsf{bc}(G)$). If in addition, we insist that bicliques must be edge-disjoint, the parameter is known as *biclique partition number* denoted by $\mathsf{bp}(G)$.

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

Proof. ($\mathsf{spd}(f) \leq \mathsf{bc}(G_f)$) Let $G = G_f$, $t = \mathsf{bc}(G)$ 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 $I_u \cap I_v \neq \emptyset$.

($\mathsf{bc}(G_f) \leq \mathsf{spd}(G_f)$) Let $G = G_f, t = \mathsf{spd}(G)$ 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 $i \in I_u$ and $i \in I_v$ implying $I_u \cap I_v \neq \emptyset$. This gives that $(u, v) \in E(G)$ from the definition of standard assignment.

($\mathsf{bp}(G_f) \leq \mathsf{uspd}(G_f)$) Let ϕ be the intersection dimension one standard assignment of ambient dimension d of f. For every $e_i \in \mathbb{F}^d$, define the set $C_i = \{(x, y) \mid \phi(x, y) = e_i\}$. We claim that $\mathcal{C} = \{C_i\}_{i \in [d]}$ is a bipartite partition of G_f. Every C_i thus defined is a biclique, because if $\phi(x, y) = e_i$ then that implies $e_i \in \phi(x)$ and $e_i \in \phi(y)$. Note that for every $(x, y) \in G_f$, there exists a unique $i \in [d]$ such that $\phi(x, y) = e_i$. 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 $\phi(x, y) = \{0\}$.

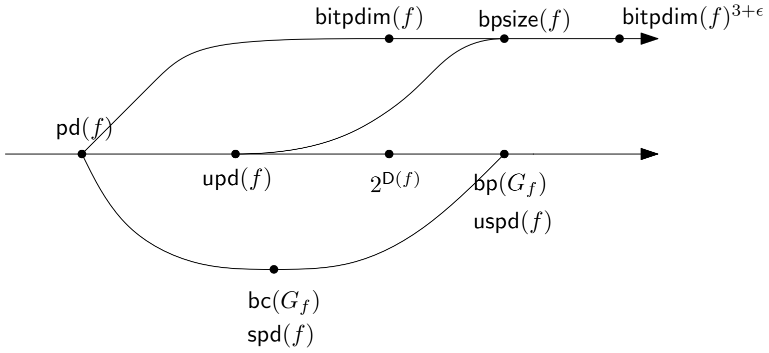
($\mathsf{uspd}(G_f) \leq \mathsf{bp}(G_f)$) Let $\mathcal{C} = \{C_i\}_{i \in [d]}$ where $d = \mathsf{bp}(G_f)$ be a biclique partition cover. We give a standard assignment ϕ for G_f defined as follows. For any x, $\phi(x) = \mathsf{span}\,\{e_i \mid \exists y, (x, y) \in C_i\}$. By definition ϕ is a standard assignment. We just need to prove that $(x, y) \in G_f$ if and only $\phi(x, y) \neq \{0\}$ and $\mathsf{dim}\phi(x, y) = 1$. 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 is exists (x, y) with $\mathsf{dim}\phi(x, y) > 1$. 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), $\mathsf{dim}\phi(x, y) \leq 1$. If $(x, y) \notin G_f$, then there does not exist an i, $(x, y) \in C_i$. But if $\phi(x, y) = e_i$ for some $i \in [d]$, then that implies by rectangle property of bicliques that $(x, y) \in C_i$, a contradiction. □

7 Discussion & 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 shown in Fig. 2.

An immediate question that arises from our work is whether $\Omega(d^2)$ lower bound on $\mathsf{upd}(\mathcal{P}_d)$ is tight. In this direction, since we have established a gap between $\mathsf{upd}(\mathcal{P}_d)$ and $\mathsf{pd}(\mathcal{P}_d)$, it is natural to study how pd and upd behave under composition of functions, in order to amplify this gap.

In another direction, we believe that the $\Omega(d^2)$ lower bound on $\mathsf{upd}(\mathcal{P}_d)$ is not tight. It is natural to study composition of functions to improve this gap.



D(f) – Deterministic Communication Complexity of f
bc(G) – Bipartite Cover number of G
bp(G) – Bipartite Partition number of G

Figure 2: Parameters considered in this work and their relations

The subspace counting based lower bounds for bitpdim that we proved are tight for functions like ED_n. However, observe that under standard complexity theoretic assumptions the bitpdim assignment for $\mathcal{P}_d$ is not tight. Hence it might be possible to use the specific linear algebraic properties of $\mathcal{P}_d$ to improve the bitpdim lower bound we obtained for $\mathcal{P}_d$.

Acknowledgements: The authors would like to thank the anonymous reviewers for several suggestions which improved the readability of the paper and specifically for pointing out that the proof of Proposition 5.1 follows from Remark 1.3 in [9]. The authors would also like to thank Noam Nisan for pointing out that the a random walk based algorithm for detecting cycles can improve the the constant in Theorem 1.2 to $3 + \epsilon$.

References

[1] Romas Aleliunas, Richard M. Karp, Richard J. Lipton, László Lovász, and Charles Rackoff. Random walks, universal traversal sequences, and the complexity of maze problems. In *20th Annual Symposium on Foundations of Computer Science, San Juan, Puerto Rico, 29-31 October 1979*, pages 218–223, 1979.

[2] E. Allender, R. Beals, and M. Ogihara. The complexity of matrix rank and feasible systems of linear equations. *Computational Complexity*, 8(2):99–126, 1999.

[3] S. Arora and B. Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, 2007.

[4] Paul Beame, Nathan Grosshans, Pierre McKenzie, and Luc Segoufin. Nondeterminism and an abstract formulation of Nečiporuk's lower bound method. *CoRR*, abs/1608.01932, 2016.

[5] Béla Bollobás. *Random Graphs, Second edition.* Cambridge Studies in Advanced Mathematics 73. Cambridge University Press, 2001.

[6] Philippe Delsarte. Association schemes and t-designs in regular semilattices. *Journal of Combinatorial Theory, Series A*, 20(2):230–243, mar 1976.

[7] Péter Frankl and Ronald L Graham. Intersection theorems for vector spaces. *European Journal of Combinatorics*, 6(2):183–187, jun 1985.

[8] Péter Frankl and Richard M Wilson. The Erdős-Ko-Rado theorem for vector spaces. *Journal of Combinatorial Theory Series A*, 43(2):228–236, nov 1986.

[9] Stasys Jukna. *Boolean Function Complexity: Advances and Frontiers*, volume 27 of *Series: Algorithms and Combinatorics.* Springer New York Inc., 2012.

[10] Ralf Koetter and Frank R. Kschischang. Coding for errors and erasures in random network coding. *IEEE Transactions on Information Theory*, 54(8):3579–3591, 2008.

[11] Eyal Kushilevitz and Noam Nisan. *Communication Complexity.* Cambridge University Press, New York, NY, USA, 1997.

[12] Satyanarayana V. Lokam. Complexity lower bounds using linear algebra. *Foundations and Trends in Theoretical Computer Science*, 4(1&2):1–155, January 2009.

[13] Benjian Lv and Kaishun Wang. The eigenvalues of q-Kneser graphs. *Discrete Mathematics*, 312(6):1144 – 1147, 2012.

[14] Michael Mitzenmacher and Eli Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis.* Cambridge University Press, New York, NY, USA, 2005.

[15] E. I. Nechiporuk. On a boolean function. *Doklady of the Academy of Sciences of the USSR*, 164(4):765–766, 1966.

[16] P. Pudlák and V. Rödl. A combinatorial approach to complexity. *Combinatorica*, 12:221–226, 1992.

[17] P. Pudlák and V. Rödl. Some combinatorial-algebraic problems from complexity theory. *Discrete Mathematics*, 136(1-3):253–279, dec 1994.

[18] Omer Reingold. Undirected connectivity in log-space. *Journal of the ACM*, 55(4):17:1–17:24, September 2008.

[19] S. Roman. *Advanced Linear Algebra.* Graduate Texts in Mathematics. Springer New York, 3rd edition, 2007.

[20] Lajos Rónyai, László Babai, and Murali K. Ganapathy. On the number of zero-patterns of a sequence of polynomials. *Journal of the AMS*, 14:2001, 2002.

[21] Heribert Vollmer. *Introduction to Circuit Complexity: A Uniform Approach.* Springer New York Inc., 1999.

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 [16], but we supply the details which are essential for the observations that we make.

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 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.

Theorem A.1. *Let $f : \{0,1\}^{2n} \to \{0,1\}$ 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, $\mathsf{pd}_{\mathbb{F}}(G_F) \leq s$*

Proof. It suffices to come up with a subspace assignment ϕ 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 $\{x_1, x_2, \dots, x_n\}$ and v corresponds to $\{x_{n+1}, \dots, x_{2n}\}$ (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 ϕ such that

$$\exists \text{ a path from start to accept in } \mathcal{B} \iff \text{Basis of } \phi(u), \phi(v) \text{ are linearly dependent} \tag{1}$$

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 $\mathcal{B}$. 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.

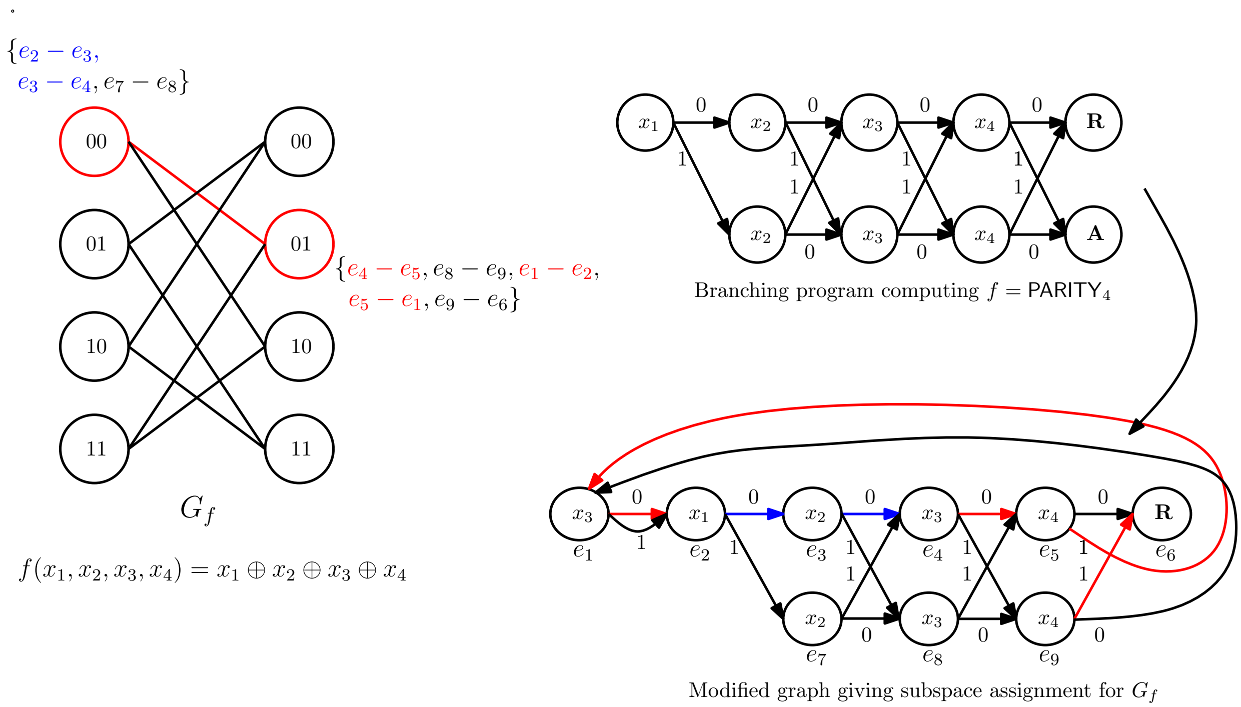
1. Modify $\mathcal{B}$ by adding a new start vertex labeled with a variable from the other partition from which v_0 got its label. For example, if V_0 is labeled with any of $x_1, x_2, \dots, x_n$, the new vertex gets its label from $\{x_{n+1}, \dots, x_{2n}\}$ and vice-versa. Connect both outgoing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

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

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 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 labeled with $\{x_{n+1}, \dots, x_{2n}\}$. Then $\phi(v)$ will be just zero subspace and $\phi(u) \cap \phi(v)$ will be trivial



Branching program computing $f = \mathsf{PARITY}_4$

Modified graph giving subspace assignment for G_f

Figure 3: Pudlák-Rödl Theorem applied to a branching program computing PARITY_4

when there is a cycle. It is to avoid this that we add a vertex labeled with variable from the other partition.

To show that ϕ is a valid subspace assignment, it remains to show that reverse implication of statement 1 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.

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

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 ϵ_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 ≥ 2 (in the undirected sense). Hence it must have an undirected cycle. □

Fig. 3 shows the transformations done to the branching program as per the proof of Pudlák-Rödl Theorem and subspace assignment obtained for 00 and 01. The intersection vector for 00 and 01 is highlighted in blue on the left partition and in red on the right partition. Notice that this intersection vector corresponds to two halves of a cycle starting from start vertex of the modified BP. The subspace assignment for each of the vertices is listed in the table below.

x_1x_2	Assignment	x_3x_4	Assignment
00	$e_2 - e_3$, $e_3 - e_4$, $e_7 - e_8$	00	$e_4 - e_5$, $e_8 - e_9$, $e_1 - e_2$, $e_5 - e_6$, $e_9 - e_1$
01	$e_2 - e_3$, $e_3 - e_8$, $e_7 - e_4$	01	$e_4 - e_5$, $e_8 - e_9$, $e_1 - e_2$, $e_5 - e_1$, $e_9 - e_6$
10	$e_2 - e_7$, $e_3 - e_4$, $e_7 - e_8$	10	$e_4 - e_9$, $e_8 - e_5$, $e_1 - e_2$, $e_5 - e_6$, $e_9 - e_1$
11	$e_2 - e_7$, $e_3 - e_8$, $e_7 - e_4$	11	$e_4 - e_9$, $e_8 - e_5$, $e_1 - e_2$, $e_5 - e_1$, $e_9 - e_6$

Table 1: Subspace assignment for PARITY_4 given by proof of Pudlák-Rödl theorem

B Bounds on the Gaussian Coefficients

Proposition B.1 (Lemma 1 of [10])**.** *For integers, $k \geq 0, n \geq k$.*

$$q^{k(n-k)} \leq \begin{bmatrix} n \\ k \end{bmatrix}_q < c_q q^{k(n-k)} \tag{2}$$

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

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,

$$\begin{aligned} \begin{bmatrix} n \\ k \end{bmatrix}_q &= \frac{(q^n-1)(q^n-q)\ldots(q^n-q^{k-1})}{(q^k-1)(q^k-q)\ldots(q^k-q^{k-1})} \\ &= \frac{q^{nk}}{q^{k^2}} \left[\frac{(1-q^{-n})(1-q^{-(n-1)})\ldots(1-q^{-(n-k+1)})}{(1-q^{-1})(1-q^{-2})\ldots(1-q^{-(k-1)})} \right] \end{aligned}$$

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 $2c_q \sum_{i=0}^{n/2} q^{in} \leq 2c_2 q^{(n^2+n)/2}$.

C Proof of Proposition 2.5

Proof. For the reverse direction, suppose there is a non zero vector w_1 in $U_1 \cap V_1$ and a non zero vector w_2 in $U_2 \cap V_2$, then $w_1^T w_2 \in U_1 \otimes U_2$ and $w_1^T w_2 \in V_1 \otimes V_2$. Hence $w = w_1^T w_2 \in (U_1 \otimes U_2) \cap (V_1 \otimes V_2)$.

For the forward direction, let w be a non zero vector in $(U_1 \otimes U_2) \cap (V_1 \otimes V_2)$. Let $\{e_i\}_{i \in [k_1]}$ be the set of basis vectors for F^{k_1} and $\{\tilde{e}_j\}_{j \in [k_2]}$ be the set of basis vectors for F^{k_2}. Hence for some $\lambda_{ij}, \mu_{ij} \in \mathbb{F}$, w can be written as, $w = \sum_{i,j} \lambda_{ij} e_i^T \tilde{e}_j = \sum_{i,j} \mu_{ij} e_i^T \tilde{e}_j$. Hence, $\sum_{i,j} (\lambda_{ij} - \mu_{ij}) e_i^T \tilde{e}_j = 0$. By linear independence of tensor basis,

$$\lambda_{ij} = \mu_{ij} \ \forall \ (i,j) \in [k_1] \times [k_2] \tag{3}$$

Since w is non-zero, there exists i_1, j_1 with $(i_1, j_1) \in [k_1] \times [k_2]$ such that $\lambda_{i_1 j_1} \neq 0$. Applying equation 3, we get $\mu_{i_1 j_1} \neq 0$. Hence for (i_1, j_1), $\lambda_{i_1 j_1}, \mu_{i_1 j_1}$ are both non-zero.

Hence it must be that $(U_1 \otimes U_2)$ and $(V_1 \otimes V_2)$ has the vector $e_{i_1}^T \tilde{e}_{j_1}$. So e_{i_1} must be present in U_1 and V_1 and e_{j_1} must be present in U_2 and V_2 (if not, $e_{i_1}^T \tilde{e}_{j_1}$ would not have appeared in the intersection). Hence $U_1 \cap V_1 \neq \{0\}$ and $U_2 \cap V_2 \neq \{0\}$. □