Constructing Fewer Open Cells by GCD Computation in CAD Projection

Jingjun Han  
School of Mathematical Sciences & Beijing International Center for Mathematical Research  
Peking University  
hanjingjunfdzf@gmail.com

Liyun Dai  
School of Mathematical Sciences & Beijing International Center for Mathematical Research  
Peking University  
dailiyun@pku.edu.cn

Bican Xia  
LMAM & School of Mathematical Sciences  
Peking University  
xbc@math.pku.edu.cn

ABSTRACT
A new projection operator of cylindrical algebraic decomposition (CAD) is proposed. The new operator computes the intersection of projection factor sets produced by different CAD projection orders. In other words, it computes the gcd of projection polynomials in the same variables produced by different CAD projection orders. We prove that the new operator still guarantees obtaining at least one sample point from every connected component of the highest dimension. In general, the gcd computation produces smaller projection factor sets and thus fewer open cells. Some examples that are difficult to be solved by existing tools have been worked out efficiently by our program based on the new operator.

Categories and Subject Descriptors
G.4 [Mathematics of computation]: Mathematical software — Algorithm design and analysis

General Terms
Algorithms

Keywords
CAD projection, positive semi-definite, polynomial.

1. INTRODUCTION
The cylindrical algebraic decomposition (CAD) method and its application to quantifier elimination (QE) for elementary real algebra was first proposed by Collins [3, 2].

A key role in CAD algorithm is its projection operator. A well known improvement of CAD projection is Hong’s projection operator which is applicable for all cases of QE [8]. For many problems of QE, a smaller projection operator given by McCallum in [9, 10], with an improvement by Brown in [1], is more efficient.

Strzeboński proposed in [2], an algorithm called generic cylindrical algebraic decomposition (GCAD) for solving systems of strict polynomial inequalities, which made use of the so-called generic projection, the same projection operator as that in [1] proposed later. Rong proposed in [14] the open CAD method which is similar to GCAD. As a property of open CAD, in terms of the Brown projection, at least one sample point can be taken from every highest dimensional cell via the open CAD lifting phase.

For traditional CAD projection operators, such as Brown’s projection operator, different projection orders may lead to a great difference in complexity. Thus, in order to reduce the projection scale, it is meaningful to study the relationship between those different projection orders. For related work, see for example [4].

The reason for such difference is mainly because the projection factors in the same variables produced by different projection orders may be different. For example, when we apply Brown’s projection operator \( B_p \) (see Definition [8]) to any given polynomial \( f \in \mathbb{Z}[x_1, ..., x_n] \), it is quite possible that
\[
B_p(f, [x_n, x_{n-1}]) \neq B_p(f, [x_{n-1}, x_n]).
\]

In this paper, we propose a new CAD projection operator \( H_p \). The new operator computes the intersection of projection factor sets produced by different CAD projection orders. In other words, it computes the gcd of projection polynomials in the same variables produced by different CAD projection orders. In some sense, the polynomial in the projection factor sets of \( H_p \) is irrelevant to the projection orders. We prove that the new operator still guarantees obtaining at least one sample point from every connected component of the highest dimension. In general, the gcd computation produces smaller projection factor sets and thus fewer open cells.

The structure of this paper is as follows. In Section 2, a simple example illustrates the main idea and steps of the new projection operator \( H_p \). Section 3 introduces basic definitions, lemmas and concepts of CAD. In Section 4, the new projection operator \( H_p \) is defined and a new algorithm based on \( H_p \) is proposed. Our main result (Theorem [3]) is proved. In Section 5, we prove that it is valid if we replace \( B_p \) with \( H_p \) in some steps of the projection phase of the simplified CAD projection \( N_p \) we proposed recently for inequality proving in...
Section 6 includes several examples which demonstrate the effectiveness of our algorithms. We conclude the paper in Section 7 with some discussions on our future work.

2. MAIN IDEA

Let us show the comparison of our new operator and Brown’s projection operator on the following simple example. Formal description and proofs of our main results are given in subsequent sections.

**Example 1.** Let \( f = x^4 - 2x^2y^2 + 2x^2z^2 + y^4 - 2y^2z^2 + z^4 + 2x^2 + 2y^2 - 4z^2 - 4 \in \mathbb{R}[x, y, z] \).

We first compute an open CAD (see Definition 4) defined by \( f \neq 0 \) in \( \mathbb{R}^3 \) by Brown’s operator. Take the order \( z \succ y \succ x \).

**Step 1**, compute the projection polynomial (up to a nonzero constant)

\[
 f_z = \text{Res}(\text{sqrf}(f), \frac{\partial}{\partial z}\text{sqrf}(f), z) = (x^4 - 2x^2y^2 + y^4 + 2x^2 + 2y^2 - 4)(3x^2 - y^2 - 4)^2
\]

where “Res” means the Sylvester resultant and “sqrf” means “squarefree” that is defined in Definition 2.

**Step 2**, compute the projection polynomial (up to a nonzero constant)

\[
 f_{z_y} = \text{Res}(\text{sqrf}(f_z), \frac{\partial}{\partial y}\text{sqrf}(f_z), y) = (3x^2 - 4)(x^4 + 2x^2 - 4)(4x^2 - 5)^2(x - 1)^8(x + 1)^8
\]

which has 8 distinct real zeros.

**Step 3**, by open CAD lifting under the order \( z \succ y \succ x \) and using the projection factor set \( \{f_{z_y}, f_z, f\} \), we will finally get 113 sample points of \( f \neq 0 \) in \( \mathbb{R}^3 \).

Now, we compute a reduced open CAD (see Definition 4) defined by \( f \in \mathbb{R}^3 \) by the new projection operator proposed in this paper. **Step 1**, take the order \( z \succ y \succ x \) and compute the projection polynomial \( f_{z_y} \) as above. **Step 2**, take another order \( y \succ z \succ x \) and we can similarly obtain a projection polynomial (up to a constant)

\[
 f_{y_z} = (3x^2 - 4)(x^4 + 2x^2 - 4)(4x^2 - 5)(6x^2 - 7)^8
\]

**Step 3**, compute

\[
 \gcd(f_{y_z}, f_{z_y}) = (3x^2 - 4)(x^4 + 2x^2 - 4)(4x^2 - 5)
\]

which has 6 distinct real zeros.

**Step 4**, by open CAD lifting under the order \( z \succ y \succ x \) and using the projection factor set \( \{\gcd(f_{y_z}, f_{z_y}), f_z, f\} \), we will finally get 87 sample points of \( f \neq 0 \) in \( \mathbb{R}^3 \).

**Remark 1.** The main result of this paper is Theorem 5 which guarantees that the new projection operator can obtain at least one sample point from every connected component of the highest dimension. Especially for Example 1 that means the intersection of the set of those 87 sample points and each connected component of \( f \neq 0 \) in \( \mathbb{R}^3 \) is not empty.

**Remark 2.** Computing projection factor sets (polynomials) under different projection orders brings additional costs compared to traditional CAD projection operators. However, it has two gains. First, it produces fewer sample points (representing open cells) in many cases as shown in Example 1. Second, the most important thing is, if the number of variables is greater than 3, it may also reduce the scale of projection. Please see Definition 13, Algorithm 3, Remark 5 and Remark 7 for details.

For problems with many variables and/or special structures, the new method performs very well. Our program can solve problems of proving polynomial inequalities with more than 10 variables (even for polynomials with 17 variables!). Please see Section 9 for details.

3. PRELIMINARIES

If not specified, for a positive integer \( n \), let \( \mathbf{x}_n \) be the set of variable \( \{x_1, \ldots, x_n\} \) and \( \alpha_n \) and \( \beta_n \) denote the point \( (\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n \) and \( (\beta_1, \ldots, \beta_n) \in \mathbb{R}^n \), respectively.

**Definition 1.** Let \( f \in \mathbb{Z}[\mathbf{x}_n] \), denote by \( \text{lc}(f, x_i) \) and \( \text{discrim}(f, x_i) \) the leading coefficient and the discriminant of \( f \) with respect to (w.r.t.) \( x_i \), respectively.

**Definition 2.** Let \( f \in \mathbb{Z}[\mathbf{x}_n] \), the set of real zeros of \( f \) is denoted by \( \text{Zero}(f) \). Denote by \( \text{Zero}(L) \) or \( \text{Zero}(f_1, \ldots, f_m) \) the common real zeros of \( L = \{f_1, \ldots, f_m\} \subset \mathbb{Z}[\mathbf{x}_n] \).

**Definition 3.** The level for \( f \in \mathbb{Z}[\mathbf{x}_n] \) is the biggest \( j \) such that (s.t.) \( \deg(f, x_j) > 0 \) where \( \deg(f, x_j) \) is the degree of \( f \) w.r.t. \( x_j \). For polynomial set \( L \subseteq \mathbb{Z}[\mathbf{x}_n] \), \( L^j \) is the set of polynomials in \( L \) with level \( i \).

**Definition 4.** Let \( P_n \) be the symmetric permutation group of \( x_1, \ldots, x_n \). Define \( P_{n,i} \) to be the subgroup of \( P_n \), where any element \( \sigma \in P_{n,i} \) fixes \( x_1, \ldots, x_{i-1}, i.e., \sigma(x_j) = x_j \) for \( j = 1, \ldots, i-1 \).

**Definition 5.** If \( h \in \mathbb{Z}[\mathbf{x}_n] \) can be factorized in \( \mathbb{Z}[\mathbf{x}_n] \) as:

\[
h = a_{t_1}^{i_1} \cdots t_1^{i_1} h_1^{i_1} \cdots h_m^{i_m},
\]

where \( a \in \mathbb{Z} \), \( t_\geq 0, m \geq 0 \), \( l_i(i = 1, \ldots, t) \) and \( h_i(i = 1, \ldots, m) \) are pairwise different irreducible primitive polynomials with positive leading coefficient (under a suitable ordering) and positive degree in \( \mathbb{Z}[\mathbf{x}_n] \). Define

\[
\text{sqrf}(h) = l_1 \cdots l_t h_1 \cdots h_m,
\]

\[
\text{sqrf}_1(h) = \{l_i, i = 1, 2, \ldots, t\},
\]

\[
\text{sqrf}_2(h) = \{h_i, i = 1, 2, \ldots, m\}.
\]

If \( h \) is a constant, let \( \text{sqrf}(h) = 1 \), \( \text{sqrf}_1(h) = \{1\}, \text{sqrf}_2(h) = \{1\} \).

In the following, we introduce some basic concepts and results of CAD. The reader is referred to [1] for a detailed discussion on the properties of CAD and open CAD.
Definition 6. An $n$-variate polynomial $f(x_{n-1},x_n)$ over the reals is said to be delineable on a subset $S$ (usually connected) of $\mathbb{R}^{n-1}$ if (1) the portion of the real variety of $f$ that lies in the cylinder $S \times \mathbb{R}$ over $S$ consists of the union of the graphs of some $t \geq 0$ continuous functions $\theta_1 \leq \cdots \leq \theta_t$ from $S$ to $\mathbb{R}$; and (2) there exist integers $m_1,\ldots,m_t \geq 1$ s.t. for every $a \in S$, the multiplicity of the root $\theta_i(a)$ of $f(a,x_n)$ (considered as a polynomial in $x_n$ alone) is $m_i$.

Definition 7. In the above definition, the $\theta_i$ are called the real root functions of $f$ on $S$, the graphs of the $\theta_i$ are called the $f$-sections over $S$, and the regions between successive $f$-sections are called $f$-sectors.

Theorem 1. Let $f(x_n,x_{n+1})$ be a polynomial in $\mathbb{Z}[x_n,x_{n+1}]$ of positive degree and $\text{disc}(f,x_{n+1})$ is a nonzero polynomial. Let $S$ be a connected submanifold of $\mathbb{R}^n$ on which $f$ is degree-invariant and does not vanish identically, and in which $\text{disc}(f,x_{n+1})$ is order-invariant. Then $f$ is analytic delineable on $S$ and is order-invariant in each $f$-section over $S$.

Based on this theorem, McCallum proposed the projection operator $\text{MCproj}$, which consists of the discriminant of $f$ and all coefficients of $f$.

Theorem 2. Let $f(x_n,x_{n+1})$ be a $(n+1)$-variate polynomial of positive degree in $x_{n+1}$ such that $\text{disc}(f,x_{n+1}) \neq 0$. Let $S$ be a connected submanifold of $\mathbb{R}^n$ in which $\text{disc}(f,x_{n+1})$ is order-invariant, the leading coefficient of $f$ is sign-invariant, and such that $f$ vanishes identically at no point in $S$. $f$ is degree-invariant on $S$.

Based on this theorem, Brown obtained a reduced McCallum projection in which only leading coefficients, discriminants and resultants appear. The Brown projection operator is defined as follows.

Definition 8. Given a polynomial $f \in \mathbb{Z}[x_n]$, if $f$ is with level $n$, the Brown projection operator for $f$ is

$$\text{Bp}(f,[x_n]) = \text{Res}(\text{sqrf}(f), \frac{\partial (\text{sqrf}(f))}{\partial x_n}, x_n).$$

Otherwise $\text{Bp}(f,[x_n]) = f$. If $L$ is a polynomial set with level $n$, then

$$\text{Bp}(L,[x_n]) = \bigcup_{f \in L} \{ \text{Res}(\text{sqrf}(f), \frac{\partial (\text{sqrf}(f))}{\partial x_n}, x_n) \} \bigcup_{f,g \in L, f \neq g} \{ \text{Res}(\text{sqrf}(f), \text{sqrf}(g), x_n) \}.$$

Define

$$\text{Bp}(f,[x_n,x_{n-1},\ldots,x_i]) = \text{Bp}(\text{Bp}(f,[x_n,x_{n-1},\ldots,x_{i+1}]),[x_i]).$$

Open CAD is a modified CAD construction algorithm, which was named in Rong Xiao’s Ph.D. thesis [14]. In fact, open CAD is similar to the generic cylindrical algebraic decomposition (GCAD) proposed in [12] and was used in DISCOVERER [13] for real root classification. For convenience, we describe the framework of the open CAD here.

Definition 9. (Open CAD) For a polynomial $f(x_n) \in \mathbb{Z}[x_n]$, an open CAD defined by $f(x_n)$ is a set of sample points in $\mathbb{R}^n$ obtained through the following three phases:

1. Projection. Use the Brown projection operator on $f(x_n)$,

   $$(1) \quad \text{Proj}(f) = \{ f, \text{Bp}(f,[x_n]), \ldots, \text{Bp}(f,[x_n,\ldots,x_2]) \};$$

2. Base. Choose one point in each of the open intervals defined by the real roots of $F^1$;

3. Lifting. Substitute each sample point of $\mathbb{R}^{i-1}$ for $x_{i-1}$ in $F^i$ and then, by the same method as Base phase, choose sample points for $F^i(x_i)$.

4. REDUCED OPEN CAD

Definition 10. (Open sample) A set of sample points $T \subset \mathbb{R}^k$ is said to be an open sample defined by $f(x_k) \in \mathbb{Z}[x_k]$ in $\mathbb{R}^k$ if it has the following property: for every open connected set $U \subset \mathbb{R}^k$ defined by $f \neq 0$, $T \cap U \neq \emptyset$.

Suppose $g(x_k)$ is another polynomial. If $T$ is an open sample defined by $f(x_k)$ in $\mathbb{R}^k$ such that $g(\alpha) \neq 0$ for any $\alpha \in T$, then we denote the open sample by $T_{g \neq 0}$.

Starting from an open sample of $\mathbb{R}^j (j < n)$, by a similar construction as the lifting phase of open CAD we can obtain a set of sample points of $\mathbb{R}^n$. More concretely, we state the procedure as follows and will prove in this section that the output of the procedure is indeed an open sample of $\mathbb{R}^n$ under some conditions.

Given two polynomial sets $L_1 = \{ f_1(x_n), f_{j-1}(x_n), \ldots, f_j(x_j) \}$ and $L_2 = \{ g_0(x_n), g_{n-1}(x_n), \ldots, g_j(x_j) \}$ and an open sample $T_{g_j \neq 0}$ defined by $f_j(x_j)$ in $\mathbb{R}^j$, we compute a set of sample points in $\mathbb{R}^n$ by repeating the following step: substitute each sample point of $\mathbb{R}^{l-1}$ for $x_{i-1}$ in $f_i (j < i \leq n)$ and then choose one point in each of the open intervals defined by the real roots of $f_i$ such that $g_i$ does not vanish at the point.

Remark 3. In the following, for convenience, we call the above procedure OpenSP and the calling sequence for inputs $L_1, L_2$ and $T$ is

$$\text{OpenSP}(L_1, L_2, T).$$

Definition 11. (Open delineable) Let

$$L_1 = \{ f_1(x_n), f_{n-1}(x_n), \ldots, f_j(x_j) \}, \quad (1)$$

$$L_2 = \{ g_0(x_n), g_{n-1}(x_n), \ldots, g_j(x_j) \} \quad (2)$$

be two polynomial sets and $S$ an open set of $\mathbb{R}^s (s \leq j)$. The polynomial $f_n(x_n)$ is said to be open delineable on $S$ w.r.t. $L_1$ and $L_2$, if for any open sample $T_{g_j \neq 0}$ defined by $f_j(x_j)$ in $\mathbb{R}^j$, any $A = \text{OpenSP}(L_1, L_2, T)$ and any open connected set $U \subset \mathbb{R}^n$ defined by $f_n \neq 0$ with $U \cap (S \times \mathbb{R}^{n-s}) \neq \emptyset$, $A \cap U \neq \emptyset$. 
Open delineability has the following four properties. The first three can be derived easily from the definition of open delineable, we will omit the proof here.

**Proposition 1. (open sample property)** Let $L_1, L_2$ be as in [1] and [3]. If $f_n(x_n)$ is open delineable on every open connected set of $f_j(x_j) \neq 0$ w.r.t. $L_1$ and $L_2$, then for any open sample $T_{g_j \neq 0}$ defined by $f_j(x_j)$ in $\mathbb{R}^2$, any $A = \text{OpenSP}(L_1, L_2, T)$ is an open sample defined by $f_n(x_n)$ in \( \mathbb{R}^n \).

**Proposition 2. (transitive property)** Let $L_1, L_2$ be as in [1] and [3]. For a given open set $S \subset \mathbb{R}^2 (s \leq j)$, there exists $k(j \leq k \leq n)$ such that $f_k(x_k)$ is open delineable on $S$ w.r.t. $\{f_k(x_k), \ldots, f_j(x_j)\}$ and $g_k(x_k)$, and $f_n(x_n)$ is open delineable on every open connected set of $f_k(x_k) \neq 0$ w.r.t. $\{f_n(x_n), \ldots, f_j(x_j)\}$ and $g_k(x_k)$, then $f_n(x_n)$ is open delineable on $S$ w.r.t. $L_1$ and $L_2$.

**Proposition 3. (nonempty intersection property)** Let $L_1, L_2$ be as in [1] and [3]. For two open sets $S_1$ and $S_2$ of $\mathbb{R}^2 (s \leq j)$ with $S_1 \cap S_2 \neq \emptyset$, if $f_n(x_n)$ is open delineable on both $S_1$ and $S_2$ w.r.t. $L_1$ and $L_2$, $f_n(x_n)$ is open delineable on $S_1 \cup S_2$ w.r.t. $L_1$ and $L_2$.

**Proposition 4. (union property)** Let $L_1, L_2$ be as in [1] and [3]. For a given open set $S \subset \mathbb{R}^2 (s \leq j)$, there exists $k(j \leq k \leq n)$ such that $f_k(x_k)$ is open delineable on $S$ w.r.t. $\{f_k(x_k), \ldots, f_j(x_j)\}$ and $g_k(x_k)$, and $f_n(x_n)$ is open delineable on every open connected set of $f_k(x_k) \neq 0$ w.r.t. $\{f_n(x_n), \ldots, f_j(x_j)\}$ and $g_k(x_k)$, then $f_n(x_n)$ is open delineable on $S_1 \cup S_2$ w.r.t. $L_1$ and $L_2$.

**Proof.** For point $\alpha_1 \in S_1, \alpha_2 \in S_2,$ s.t. $g_j(\alpha_1)g_j(\alpha_2) \neq 0, t = 0, 1, 2,$ let $T_{g_j \neq 0}^{(i \neq 0)}$ be any open sample defined by $f_j$ with $T_{g_j \neq 0}^{(i \neq 0)} \cap S_i = \{\alpha_i\}$. For any $A' = \text{OpenSP}(L_1, L_2, T_{g_j \neq 0}^{(i \neq 0)})$, let $A_{\alpha_1} = \{\beta_1 \in A' | \beta_1 \in S_1 \} = \{\alpha_1\}$. Let $T_{g_j \neq 0}^{(i \neq 0)}$ be any open sample defined by $f_j$ with $T_{g_j \neq 0}^{(i \neq 0)} \cap (S_1 \cup S_2) = \{\alpha_1\}$. For any $A' = \text{OpenSP}(L_1, L_2, T_{g_j \neq 0}^{(i \neq 0)})$, let $A_{\alpha_1} = \{\beta_1 \in A' | \beta_1 \in S_1 \} = \{\alpha_1\}$.

For any open connected set $U$ defined by $f_n \neq 0$ with $U \cap (\alpha_1 \times \mathbb{R}^{n-1}) \neq \emptyset$, then $A_{\alpha_1} \cap U \neq \emptyset$ and $A_{\alpha_2} \cap U \neq \emptyset$. Since $f_n(x_n)$ is open delineable on $S_1 \cup S_2$ w.r.t. $L_1$ and $L_2$, $A_{\alpha_1} \cap U \neq \emptyset$. This implies that $U \cap (S_2 \times \mathbb{R}^{n-1}) \neq \emptyset$. Thus, $A_{\alpha_2} \cap U \neq \emptyset$. Therefore, $f_n(x_n)$ is open delineable on $(S_1 \cup S_2) \cap \text{Zero}(g_j)$ w.r.t. $L_1$ and $L_2$. Since $0 \neq g_j(S_1 \cup S_2)$, $f_n(x_n)$ is open delineable on $S_1 \cup S_2$ w.r.t. $L_1$ and $L_2$.

Now, we define the new projection operator $\text{H}_p$.

**Definition 12.** Let $f \in \mathbb{Z}[x_1, \ldots, x_n]$. For $m(1 \leq m \leq n), \text{denote } [y] = [y_1, \ldots, y_m] \text{ where } y_i \in \{x_1, \ldots, x_n\} \text{ for } 1 \leq i \leq m \text{ and } y_i \neq y_j \text{ for } i \neq j$. For $1 \leq i \leq m, \text{H}_p(f, [y], y_i) \text{ and } \text{H}_p(f, [y]) \text{ are defined recursively as follows.}$

\[ \text{H}_p(f, [y], y_i) = \text{H}_p(\text{H}_p(f, [y]), y_i) \]
\[ \text{H}_p(f, [y]) = \gcd(\text{H}_p(f, [y], y_1), \ldots, \text{H}_p(f, [y], y_m)) \]

where $[y] = [y_1, \ldots, y_{i-1}, y_{i+1}, \ldots, y_m]$ and $\text{H}_p([]) = f$.

**Proposition 5.** Let $f \in \mathbb{Z}[x_n]$ be a squarefree polynomial with level $n$. $f$ is open delineable on every open connected set defined by $\text{H}_p(f, [x_n]) \neq 0$ in $\mathbb{R}^{n-1}$ w.r.t. $\text{H}_p(f, n)$ and $\text{H}_p(f, n)$.

**Definition 13. (Reduced open CAD)** A reduced open CAD of $f(x_n)$ w.r.t. $[x_n, \ldots, x_j]$ is a set of sample points in $\mathbb{R}^n$ obtained through the following three phases:

1. **Projection.** Compute $\text{H}_p(f, j)$ and $\text{H}_p(f, j)$;

2. **Base.** Choose an open sample $T_{\text{H}_p(f, [x_n, \ldots, x_j], x_j) \neq 0}$ defined by $\text{H}_p(f, [x_n, \ldots, x_j])$ in $\mathbb{R}^{j-1}$;

3. **Lifting.** Substitute each sample point $\beta_{j-i} \in \mathbb{R}^{i-1} (i \geq j)$ for $x_{i-1}$ in $\text{H}_p(f, [x_n, \ldots, x_{i+1}])$ and then choose one point in each of the open intervals defined by the real roots of $\text{H}_p(f, [x_n, \ldots, x_{i+1}], x_{i+1})$ and such that $\text{H}_p(f, [x_n, \ldots, x_{i+1}], x_{i+1}) (\beta_{j-i})$ does not vanish at that point.

**Lemma 1.** Let $f$ and $g$ be coprime in $\mathbb{Z}[x_n]$. For any connected open set $U$ in $\mathbb{R}^n$, the open set $V = U \setminus \text{Zero}(f, g)$ is also connected.

The following Theorem is the main result of this paper, which shows that the reduced open CAD owns the property of open delineability.

**Theorem 3.** Let $j$ be an integer and $2 \leq j \leq n$. For any given polynomial $f(x_n)$ in $\mathbb{Z}[x_n]$ and any open connected set $U \subset \mathbb{R}^{n-1} \setminus \text{H}_p(f, [x_n, \ldots, x_j])$, let $S = U \setminus \text{Zero}(\text{H}_p(f, [x_n, \ldots, x_j], x_j) \{x = j, \ldots, n\})$. Then $f(x_n)$ is open delineable on the open connected set $S$ w.r.t. $\text{H}_p(f, j)$ and $\text{H}_p(f, j)$. As a result, a reduced open CAD of $f(x_n)$ w.r.t. $[x_n, \ldots, x_j]$ is an open sample defined by $f(x_n)$.

**Proof.** First, by Lemma 1 $S$ is open connected. We prove the theorem by induction on $k = n - j$. When $k = 0,$
it is obvious true from Proposition 3. Suppose the theorem is true for all polynomials \( f(x_k) \in \mathbb{Z}[x_k] \) with \( k = 0, 1, \ldots, n - i - 1 \). We now consider the case \( k = n - i \). Let \( [a, \ldots, x_i] \). For any given polynomial \( f(x_n) \in \mathbb{Z}[x_n] \), let \( U \subset \mathbb{R}^{i-1} \) be an open connected set of \( \mathbb{H}(f, [x]) \neq 0 \) and \( S = U \setminus \text{Zero}(\mathbb{H}(f, [x], x_i) | i = 1, \ldots, n) \).

For any point \( \alpha \in S \) with \( \mathbb{H}(f, [x], x_i)(\alpha) \neq 0 \), there exists an open connected set \( S_{\alpha} \subset \mathbb{R}^{i-1} \) such that \( \alpha \in S_{\alpha} \) and \( 0 \notin \mathbb{H}(f, [x], x_i)(S_{\alpha}) \). By induction, \( \mathbb{H}(f, [x_{n-i}, \ldots, x_i]) \) is open delineable on \( S_{\alpha} \) w.r.t. \( \{\mathbb{H}(f, [x], x_i)\} \) and \( \{\mathbb{H}(f, [x], x_i)\} \). By induction again and the transitive property of open delineable (Proposition 2), \( f \) is open delineable on \( S_{\alpha} \) w.r.t. \( \mathbb{H}(f, [x], x_i) \) and \( \mathbb{H}(f, [x], x_i) \).

To summarize, the above discussion shows that for any point \( \alpha \in S \), there exists an open connected set \( S_{\alpha} \subset \mathbb{R}^{i-1} \) such that \( \alpha \in S_{\alpha} \) and \( f \) is open delineable on \( S_{\alpha} \) w.r.t. \( \mathbb{H}(f, [x], x_i) \) and \( \mathbb{H}(f, [x], x_i) \). By the nonempty intersection property of open delineable (Proposition 3) and the fact that \( S \) is connected, \( f(x_n) \) is open delineable on \( S \) w.r.t. \( \mathbb{H}(f, [x], x_i) \) and \( \mathbb{H}(f, [x], x_i) \) as desired.

Therefore, the theorem is true for \( k = n - j \) by induction.

**Remark 4.** If we do not require \( U \) to be open in Theorem 3, i.e., \( U \subset \mathbb{R}^{i-1} \) be a connected submanifold, in which \( \mathbb{H}(f, [x_{n-i}, \ldots, x_i]) \) is order-invariant. If \( S \) is connected, we may generalize the notation of open delineable and obtain similar results to Theorem 3 with similar discussion. That will make it possible to choose at least one sample point from each connected component of \( f \neq 0 \) by using \( \mathbb{H}(f, [x], x_i) \).

As an application of Theorem 3 for a given polynomial \( f(x_n) \), we could obtain a CAD based method to get an open sample defined by \( f \). Roughly speaking, if we already got an open sample defined by \( \mathbb{H}(f, [x_{n-i}, \ldots, x_i]) \) in \( \mathbb{R}^{i-1} \), according to Theorem 3, we could obtain an open sample defined by \( f \). That process could be done recursively.

**Remark 5.** Roughly speaking, in the definition of \( \mathbb{H}(f, [x], x_i) \), we first choose \( m \) variables from \( \{x_{n-i}, \ldots, x_i\} \), compute all projection polynomials under all possible orders of those \( m \) variables, and then compute the gcd of all those projection polynomials. It is not difficult to see that if we modify the definition of \( \mathbb{H}(f, [x], x_i) \) by choosing several (not all) orders of those \( m \) variables and computing the gcd of the projection polynomials under those orders, Theorem 3 is still valid.

Therefore, Theorem 3 provides us many ways for designing various algorithms for computing open samples. Obviously, different choices of \( m \) and different orderings of the \( m \) variables at each step of recursion will result in different algorithms. For example, we may set \( m = 2 \) and choose \( \{x_{n-i-1}, x_{n-i-2}, x_{n-i-3}\} \), etc. successively in each step. Because there are only two different orders for two variables, we compute the gcd of two projection polynomials under the two orders in each step. The following algorithm is based on this choice.

**Algorithm 1 \( \mathbb{H}(f, [x], x_i) \)**

**Require:** A polynomial \( f \in \mathbb{Z}[x_n] \) of level \( n \).

**Ensure:** An open sample defined by \( f \), i.e., a set of sample points which contains at least one point from each connected component of \( f \neq 0 \) in \( \mathbb{R}^{n} \).

1: \( g := f; \)
2: \( L_1 := \{\}; \)
3: \( L_2 := \{\}; \)
4: for \( i \) from \( n-1 \) downto 2 do
5: if \( i \geq 3 \) then
6: \( L_1 := L_1 \cup \mathbb{H}(g, i-1); \)
7: \( L_2 := L_2 \cup \mathbb{H}(g, i-1); \)
8: \( g := \mathbb{H}(g, [x_i, x_{i-1}]); \)
9: \( i := i - 1; \)
else
10: \( L_1 := L_1 \cup \mathbb{H}(g, i); \)
11: \( L_2 := L_2 \cup \mathbb{H}(g, i); \)
12: \( g := \mathbb{H}(g, [x_i]); \)
end if
13: end for
14: \( C := \text{OpenSP}(L_1, L_2, T); \)
15: return \( C \).

**Remark 6.** If \( \mathbb{H}(f, [x_{n-i-1}]) \neq \mathbb{H}(f, [x_{n-i-1}]) \) and \( \alpha > 3 \), it is obvious that the scale of projection in Algorithm 1 is smaller than that of open CAD in Definition 4.

**Remark 7.** Let \( f(x_1) \) and \( g(x_1) \) be two univariate polynomials. Isolating the real roots of \( f(x_1) \) and choosing one point from each open interval defined by the real roots such that \( g(x_1) \) does not vanish at that point will give an open sample \( T_{g \neq 0} \) in \( \mathbb{R} \) defined by \( f \).

**5. PROJECTION OPERATOR NP**

In this section, we combined the idea of \( \mathbb{H}(f, [x], x_i) \) and the simplified CAD projection operator \( \mathbb{H}(f, [x], x_i) \) we introduced previously in [7], to get a new algorithm for proving polynomial inequality.

**Definition 14.** Suppose \( f \in \mathbb{Z}[x_n] \) is a polynomial of level \( n \). Define
\[
\text{Oc}(f, x_n) = \text{sqrf}_{t}(\text{lc}(f, x_n)), \quad \text{Od}(f, x_n) = \text{sqrf}_{t}(\text{discrim}(f, x_n)), \\
\text{Ecs}(f, x_n) = \text{sqrf}_{t}(\text{lc}(f, x_n)), \quad \text{Ed}(f, x_n) = \text{sqrf}_{t}(\text{discrim}(f, x_n)), \\
\text{Ocd}(f, x_n) = \text{Oc}(f, x_n) \cup \text{Od}(f, x_n), \\
\text{Edc}(f, x_n) = \text{Ecs}(f, x_n) \cup \text{Ed}(f, x_n).
\]
The secondary and principal parts of the projection operator \( Np \) are defined as
\[
Np_1(f, [x_n]) = \text{Ocd}(f, x_n),
\]
\[
Np_2(f, [x_n]) = \{ \prod_{g \in \text{Ocd}(f, x_n)} g \}.
\]

If \( L \) is a set of polynomials of level \( n \), define
\[
Np_1(L, [x_n]) = \bigcup_{g \in L} \text{Ocd}(g, x_n),
\]
\[
Np_2(L, [x_n]) = \bigcup_{g \in L} \{ \prod_{h \in \text{Ocd}(g, x_n)} h \}.
\]

Based on the projection operator \( Np \), we proposed an algorithm, \textit{Proineq}, in [7] for proving polynomial inequalities. Algorithm \textit{Proineq} takes a polynomial \( f(x_n) \in \mathbb{Z}[x_n] \) as input, and returns whether or not \( f(x_n) \geq 0 \) on \( \mathbb{R}^n \). The readers are referred to [7] for the details of \textit{Proineq}.

The projection operator \( Np \) is extended and defined in the next definition.

**Definition 15.** Let \( f \in \mathbb{Z}[x_1, \ldots, x_n] \) with level \( n \). Denote \( [y] = [y_1, \ldots, y_m] \), for \( 1 \leq m \leq n \), where \( y_i \in \{x_1, \ldots, x_n\} \) for \( 1 \leq i \leq m \) and \( y_i \neq y_j \) for \( i \neq j \). Define
\[
Np(f, [x_i]) = Np_2(f, [x_i]), Np(f, [x_i], x_i) = \prod_{g \in Np(f, [x_i])} g.
\]

For \( m(m \geq 2) \) and \( i(1 \leq i \leq m) \), \( Np(f, [y], y_i) \) and \( Np(f, [y]) \) are defined recursively as follows:
\[
Np(f, [y], y_i) = \text{Bp}(Np(f, [y], y_i)),
\]
\[
Np(f, [y]) = \text{gcd}(Np(f, [y], y_1), \ldots, Np(f, [y], y_m)),
\]

where \( [y] = [y_1, \ldots, y_{i-1}, y_{i+1}, \ldots, y_m] \). Define
\[
Np(f, i) = \{ f, Np(f, [x_n]), \ldots, Np(f, [x_n, \ldots, x_i]) \},
\]
and
\[
Np(f, i) = \{ f, Np(f, [x_n], x_n), \ldots, Np(f, [x_n, \ldots, x_i]) \}.
\]

**Theorem 5.** Let \( j \) be an integer and \( 2 \leq j \leq n \). For any given polynomial \( f(x_n) \in \mathbb{Z}[x_n] \), and any open connected set \( U \) of \( \mathbb{Z}^n \), let \( S = U \backslash \text{Zero} \left( Np(f, [x_n, \ldots, x_j], x_j) \right) \). If the polynomials in \( \bigcup_{i=0}^{n-j} Np_i(f, [x_n, \ldots, x_i]) \) are all semi-definite on \( U \times \mathbb{R}^{n-j} \), \( f(x_n) \) is open semi-definite on \( S \) w.r.t. \( Np(f) \) and \( Np(f, j) \).

Theorem [5] and Proposition [3] provide us a new way to decide the non-negativity of a polynomial as stated in the next theorem.

**Theorem 6.** Given a positive integer \( n \geq 3 \). Let \( f \in \mathbb{Z}[x_n] \) be a squarefree polynomial with level \( n \) and \( U \) a connected open set of \( Np(f, [x_n, \ldots, x_j]) \neq 0 \) in \( \mathbb{R}^{n-1} \). Denote \( S = U \backslash \text{Zero}(Np(f, [x_n, \ldots, x_j], x_j)) \). The necessary and sufficient condition for \( f(x_n) \) to be positive semi-definite on \( U \times \mathbb{R}^{n-j+1} \) is the following two conditions hold.
\[
(1) \quad \text{The polynomials in } \bigcup_{i=0}^{n-j} Np_i(f, [x_n, \ldots, x_i]) \text{ are all semi-definite on } U \times \mathbb{R}^{n-j}.
\]
\[
(2) \quad \text{There exists a point } \alpha \in S \text{ such that } f(\alpha, x_j, \ldots, x_n) \text{ is positive semi-definite on } \mathbb{R}^{n-j+1}.
\]

Based on the above theorems, it is easy to design some different algorithms (depending on the choice of \( j \)) to prove polynomial inequality. For example, the heuristic algorithm \textit{PSD-NpTwo} for deciding whether a polynomial is positive semi-definite, which we will introduce later, is based on Theorem [5] when \( j = n-1 \) (Proposition [3]).

**Proposition 7.** Given a positive integer \( n \geq 3 \). Let \( f \in \mathbb{Z}[x_n] \) be a squarefree polynomial with level \( n \) and \( U \) a connected open set of \( Np(f, [x_n, x_{n-1}]) \neq 0 \) in \( \mathbb{R}^{n-2} \). Denote \( S = U \backslash \text{Zero}(Np(f, [x_n, x_{n-1}], x_{n-1})) \). The necessary and sufficient condition for \( f(x_n) \) to be positive semi-definite on \( U \times \mathbb{R}^2 \) is the following two conditions hold.
\[
(1) \quad \text{The polynomials in either } Np_1(f, [x_n]) \text{ or } Np_1(f, [x_{n-1}]) \text{ are semi-definite on } U \times \mathbb{R}.
\]
\[
(2) \quad \text{There exists a point } \alpha \in S \text{ such that } f(\alpha, x_{n-1}, x_n) \text{ is positive semi-definite on } \mathbb{R}^2.
\]

**Lemma 3.** Given a polynomial \( f(x_n, x) \in \mathbb{Z}[x_n, x] \), say
\[
f(x_n, x) = \sum_{i=0}^{l} c_i x^{i}, c_l \neq 0,
\]
where \( c_i \) is a polynomial in \( x_n \) for each \( i = 0, \ldots, l \). Let
\[
f^h(x_n, x, y) = \sum_{i=0}^{l} c_i x^{i} y^{i-h},
\]
then \( f \geq 0 \) for all \( (x_n, x) \in \mathbb{R}^{n+1} \) is equivalent to \( f^h \geq 0 \) for all \( (x_n, x, y) \in \mathbb{R}^{n+2} \).
Algorithm 2 PSD-HpTwo

Require: An irreducible polynomial \( f \in \mathbb{Z}[x_n] \).
Ensure: Whether or not \( \forall \alpha_n \in \mathbb{R}^n, f(\alpha_n) \geq 0 \).
1: if \( n \leq 2 \) then
2: return false
3: end if
4: \( L_1 := \text{Np}(f, [x_n]) \cup \text{Np}_1(f, [x_{n-1}]) \)
5: \( L_2 := \text{Np}(f, [x_n, x_{n-1}]) \)
6: for \( g \) in \( L_1 \) do
7: if \( \text{PSD-Hp}(g) = \text{false} \) then
8: return false
9: end if
10: \( \text{end for} \)
11: \( C_{n-2} := \text{A reduced open CAD of} L_2 \) w.r.t. \([x_{n-2}, \ldots, x_2] \), which satisfies that \( \text{Zero}(\text{Np}(f, [x_{n}, x_{n-1}], x_n), \text{Np}(f, [x_{n}, x_{n-1}], x_{n-1})) \) \( \cap C_{n-2} = \emptyset \).
12: if \( \exists \alpha_{n-2} \in C_{n-2} \) such that \( \text{Proineq}(f(\alpha_{n-2}, x_{n-1}, x_n)) = \text{false} \) then
13: return false
14: end if
15: end if
16: \( \text{end for} \)
17: \( \text{end if} \)
18: return true

Definition 16. Let
\( E_t(f, x) := \text{sqrf}_2(\text{lc}(f^h, x)) \cap \text{sqrf}_1(\text{lc}(f^h, y)), \)
\( O_t(f, x) := \text{sqrf}_1(\text{lc}(f^h, x)) \cap \text{sqrf}_1(\text{lc}(f^h, y)), \)
\( E_{td}(f, x) := E_t(f, x) \cup E_d(f, x), \)
\( O_{td}(f, x) := O_t(f, x) \cup O_d(f, x). \)
The projection operator \( \text{HP}_p \) is defined as
\( \text{HP}_p(f, [x]) = O_{td}(f, x), \)
\( f = ax^3 + (a + b + c)x^2 + (a^2 + b^2 + c^2)x + a^3 + b^3 + c^3 - 1. \)
Under the order \( a < b < c < x \), an open CAD defined by \( f \) has 132 sample points, while an open sample obtained by the algorithm \( \text{HP}_p \) has 15 sample points.

Example 6.2. For 100 random polynomials \( f(x, y, z) \) with degree \( 1 \), \( \text{Figure 1} \) shows the numbers of real roots of \( \text{Bp}(f, [z, y]), \text{Bp}(f, [y, z]), \) and \( \text{Bp}(f, [y, z]), \) respectively. It is clear that the number of real roots of \( \text{Bp}(f, [x, z]) \) is always less than those of \( \text{Bp}(f, [z, y]) \) and \( \text{Bp}(f, [y, z]). \)

![Figure 1: The number of real roots.](image)

Algorithm \( \text{PSD-HpTwo} \) has been implemented using Maple. Now, we illustrate the performance of our program with several non-trivial examples. We report the timings of the program \( \text{PSD-HpTwo} \), the program \( \text{Proineq} \) \( [7] \), and the program \( \text{RAGlib} \) \( [8] \) on these examples.

All computations were performed on a laptop with Inter Core2 2.10GHz CPU and 2.00GB RAM, Windows XP and Maple 15.

"Generated by \text{randpoly}((x, y, z), \text{degree}=8) \) in Maple 15.

RAGlib release 3.19.4 (Oct., 2012) implemented by Mohab Safey El Din.
For more examples, please visit the homepage[3] of the first author.

**Example 6.3.** [3] Prove that
\[
\left( \sum_{i=1}^{n} x_i \right)^2 - 4 \sum_{i=1}^{n} x_i^2 x_{i+1} \geq 0,
\]
where \(x_{n+1} = x_1\).

Hereafter “>4000” means either the running time is over 4000 seconds or the software is failure to get an answer. The timings in the table are in seconds.

| \(n\) | PSD-HpTwo | Raglib | Proineq |
|------|-----------|--------|---------|
| 5    | 0.156     | 6.984  | 0.297   |
| 8    | 0.635     | 177.750| >4000   |
| 11   | 4.688     | 3990.015| >4000   |
| 14   | 36.453    | >4000  | >4000   |
| 17   | 295.110   | >4000  | >4000   |

**Example 6.4.** Prove that
\[
B(x_{3m+2}) = \left( \sum_{i=1}^{3m+2} x_i \right)^2 - 2 \sum_{i=1}^{3m+2} x_i^2 \sum_{j=1}^{m} x_{i+j+1}^2 \geq 0,
\]
where \(x_{3m+2+r} = x_r\). When \(m = 1\), it is equivalent to the case \(n = 5\) of the last example. This form was once studied in [1].

| \(3m+2\) | PSD-HpTwo | Raglib | Proineq |
|----------|-----------|--------|---------|
| 5        | 0.156     | 6.984  | 0.297   |
| 8        | 1.000     | 144.907| 23.094  |
| 11       | 7.751     | 2989.516| >4000   |

7. CONCLUSION
In this paper, we propose a new CAD projection operator \(Hp\). The new operator computes the intersection of projection factor sets produced by different CAD projection orders. In other words, it computes the gcd of projection polynomials in the same variables produced by different CAD projection orders. In some sense, the polynomial in the projection factor sets of \(Hp\) is irrelevant to the projection orders. We prove that the new operator still guarantees obtaining at least one sample point from every connected component of the highest dimension. Some examples that are difficult to be solved by existing tools have been worked out efficiently by our program based on the new operator.

It is not difficult to see that, if the input polynomial \(f(x_n)\) is symmetric, the new projection operator \(Hp\) cannot reduce the projection scale and the number of sample points. If \(f(x_n)\) is not symmetric, it is most likely that \(Hp\) produces smaller projection factor sets and thus fewer open cells. On the other hand, as stated in Remark[4] one has many choices based on Theorem[3] to design different algorithms for computing open samples defined by \(f(x_n)\). How to get better strategies of choosing variables and orders for concrete examples is our ongoing work.

8. ACKNOWLEDGEMENTS
The authors would like to convey their gratitude to Hoon Hong who provided his valuable comments, advice and suggestion on this paper when he visited Peking University. The authors also would like to thank all members in the symbolic computation seminar of Peking University, including Xiaoxian Tang, Zhenghong Chen and Ting Gan, who together checked the proofs.

9. REFERENCES
[1] Brown C W. Improved projection for cylindrical algebraic decomposition. J.Symbolic Computation, 2001:32: 447–465.
[2] Caviness B F, Johnson J R. Quantifier elimination and cylindrical algebraic decomposition. Springer Verlag, 1998.
[3] Collins, G. E. Quantifier Elimination for the Elementary Theory of Real Closed Fields by Cylindrical Algebraic Decomposition. Lect. Notes Comput. Sci. 33, 134–183, 1975.
[4] Dolzmann A, Seidl A, Sturm T. Efficient projection orders for CAD. Proceedings of the 2004 international symposium on Symbolic and algebraic computation. ACM, 2004: 111-118.
[5] M. S. El Din, RAGLib (Real Algebraic Geometry Library), 2007.
[6] Han J J: An Introduction to the Proving of Elementary Inequalities. Harbin: Harbin Institute of Technology Press, 2011. 49-50. (in Chinese)
[7] Han J J, Jin Z, and Xia B C. Proving Inequalities and Solving Global Optimization Problems via Simplified CAD Projection, arXiv preprint [arXiv:1205.1223], 2012.
[8] Hong H. An improvement of the projection operator in cylindrical algebraic decomposition. In: Proceedings of the International Symposium on Symbolic and Algebraic Computation, 261–264, 1990.
[9] McCallum, S. An improved projection operation for cylindrical algebraic decomposition of three-dimensional space. J.Symbolic Computation., 1988, 5:141–161.
[10] McCallum, S. An improved projection operation for cylindrical algebraic decomposition. In Caviness, B., Johnson, J. eds, Quantifier Elimination and Cylindrical Algebraic Decomposition. Texts and Monographs in Symbolic Computation. Vienna, Springer-Verlag, 242–268, 1998.
[11] Parrilo, P.A., 2000. Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization. Dissertation (Ph.D.), California Institute of Technology.
[12] Strzeboński, A. Solving systems of strict polynomial inequalities. J. Symbolic Comput. 29, 471–480, 2000.
[13] Xia B, DISCOVERER: A tool for solving problems involving polynomial inequalities. In: Proceedings of ATCM’2000, 472–481, ATCM Inc., Blacksburg, USA, Dec, 2000.
[14] Xiao R, Parametric Polynomial Systems Solving. Dissertation (Ph.D.), Peking University, 2009. (in Chinese)