Metric Dimension of Hamming Graphs and Applications to Computational Biology

Lucas Laird

Department of Computer Science and Department of Applied Math
University of Colorado, Boulder

July 1, 2020

Introduction

K-mer: A string of length k with symbols chosen from a reference alphabet.

Introduction

K-mer: A string of length k with symbols chosen from a reference alphabet.

Examples:

- Genetic sequences: A,C,T,G base pairs
- Amino Acid sequences: 20 different amino acids
- Phone numbers: digits

Introduction

K-mer: A string of length k with symbols chosen from a reference alphabet.

Examples:

- Genetic sequences: A,C,T,G base pairs
- Amino Acid sequences: 20 different amino acids
- Phone numbers: digits

The **Hamming distance** between two k-mers, denoted $d(\cdot, \cdot)$, is the number of positions where they differ.

Introduction

K-mer: A string of length k with symbols chosen from a reference alphabet.

Examples:

- Genetic sequences: A,C,T,G base pairs
- Amino Acid sequences: 20 different amino acids
- Phone numbers: digits

The **Hamming distance** between two k-mers, denoted $d(\cdot, \cdot)$, is the number of positions where they differ.

Example: $d(ABBA, ACBC) = 2$ since they differ at two positions.

Introduction

Hamming graph

A **Hamming graph** $H_{k,a}$ has a vertex set of all k-mers made from an alphabet of size a. Two vertices v_1, v_2 are **neighbors** if $d(v_1, v_2) = 1$. The shortest path distance between any two vertices is their Hamming distance.

A **Hamming graph** $H_{k,a}$ has a vertex set of all k-mers made from an alphabet of size a. Two vertices v_1, v_2 are **neighbors** if $d(v_1, v_2) = 1$. The shortest path distance between any two vertices is their Hamming distance.

Example:

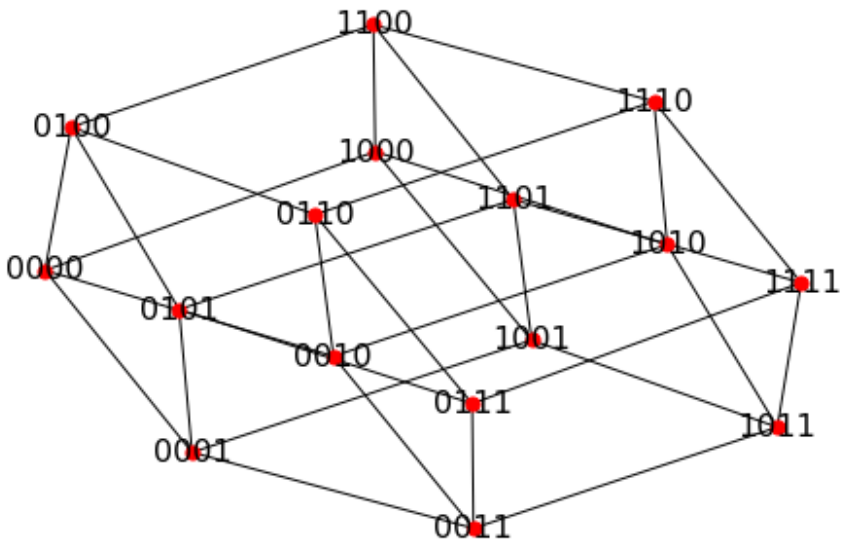

Figure: Hamming graph $H_{k=4,a=2}$

Metric Dimension and Resolving Sets

Definition (Resolving Set [1, 2])

Let $G = (V, E)$ be a connected simple graph and $d(x, y)$ be the shortest path distance between vertices x, y in G. A set $R \subseteq V$ is **resolving** if for every pair of distinct vertices $v_1, v_2 \in V$ there is an $r \in R$ such that $d(v_1, r) \neq d(v_2, r)$.

Metric Dimension and Resolving Sets

Definition (Resolving Set [1, 2])

Let $G = (V, E)$ be a connected simple graph and $d(x, y)$ be the shortest path distance between vertices x, y in G. A set $R \subseteq V$ is **resolving** if for every pair of distinct vertices $v_1, v_2 \in V$ there is an $r \in R$ such that $d(v_1, r) \neq d(v_2, r)$.

The **metric dimension**, $\beta(G)$, is the size of a smallest possible resolving set.

Resolving Set Embeddings

The vertices of a graph can be embedded by using a resolving set.

Resolving Set Embeddings

The vertices of a graph can be embedded by using a resolving set.

Example:

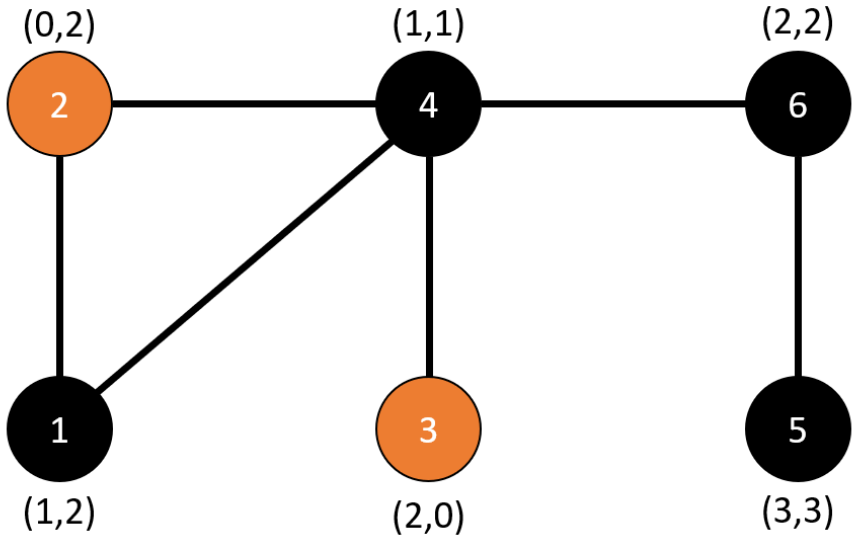

Figure: $R = \{2, 3\}$ is a minimal resolving set

Resolving Set Embeddings

The vertices of a graph can be embedded by using a resolving set.

Example:

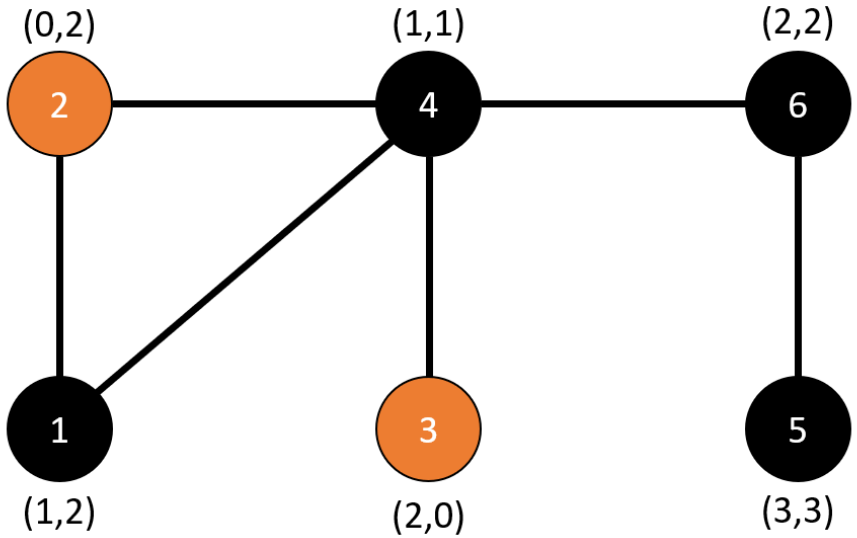

Figure: $R = \{2, 3\}$ is a minimal resolving set

- The size of the resolving set is the dimension of the embedding.

Motivating Example

What if we wanted to check if any nodes in this resolving set on $H_{8,20}$ [3] can be removed so that it stays resolving?

$$\left\{\begin{array}{cccccc}
AAAAAAAA, & AAAAAAAR, & AAAAAARA, & AAAAARAA, & AAAARAAA, & AAARAAAA, \\
ARWAAAAA, & CCCHHHHH, & CCCHHHHI, & CCCHHHIA, & CCCHHIAA, & CCCHIAAA, \\
CCCIAAAA, & CNSAAAAA, & DDDEEEEE, & DDDEEEEG, & DDDEEEGA, & DDDEEGAA, \\
DDDEGAAA, & DDDGAAAA, & DHFAAAAA, & EAGAAAAA, & EEEFAAAA, & EEEMFAAA, \\
EEEMMFAA, & EEEMMMFA, & EEEMMMMF, & EEEMMMMM, & FFFAAAAA, & GGGPPPPP, \\
GGGPPPPS, & GGGPPPSA, & GGGPPSAA, & GGGPSAAA, & GGGSAAAA, & HHHTTTTT, \\
HHHTTTTW, & HHHTTTWA, & HHHTTWAA, & HHHTWAAA, & HHHWAAAA, & HPVAAAAA, \\
IIIVAAAA, & IIIYVAAA, & IIIYYVAA, & IIIYYYVA, & IIIYYYYV, & IIIYYYYY, \\
KKKAAAAA, & KLQAAAAA, & LLLAAAAA, & MKYAAAAA, & MMMMAAAA, & NNNCCCCC, \\
NNNCCCCQ, & NNNCCCQA, & NNNCCQAA, & NNNCQAAA, & NNNQAAAA, & NSTAAAAA, \\
PPPAAAAA, & QPKAAAAA, & QQQKAAAA, & QQQLKAAA, & QQQLLKAA, & QQQLLLKA, \\
QQQLLLLK, & QQQLLLLL, & QYEAAAAA, & RRRDAAAA, & RRRNDAAA, & RRRNNDAA, \\
RRRNNNDA, & RRRNNNND, & RRRNNNNN, & SISAAAAA, & SVTAAAAA, & TTCAAAAA, \\
VFRAAAAA, & WMPAAAAA, & WWDAAAAA, & YGLAAAAA & &
\end{array}\right\}$$

Motivating Example

What if we wanted to check if any nodes in this resolving set on $H_{8,20}$ [3] can be removed so that it stays resolving?

$$\left\{\begin{array}{cccccc}
AAAAAAAA, & AAAAAAAR, & AAAAAARA, & AAAAARAA, & AAAARAAA, & AAARAAAA, \\
ARWAAAAA, & CCCHHHHH, & CCCHHHHI, & CCCHHHIA, & CCCHHIAA, & CCCHIAAA, \\
CCCIAAAA, & CNSAAAAA, & DDDEEEEE, & DDDEEEEG, & DDDEEEGA, & DDDEEGAA, \\
DDDEGAAA, & DDDGAAAA, & DHFAAAAA, & EAGAAAAA, & EEEFAAAA, & EEEMFAAA, \\
EEEMMFAA, & EEEMMMFA, & EEEMMMMF, & EEEMMMMM, & FFFAAAAA, & GGGPPPPP, \\
GGGPPPPS, & GGGPPPSA, & GGGPPSAA, & GGGPSAAA, & GGGSAAAA, & HHHTTTTT, \\
HHHTTTTW, & HHHTTTWA, & HHHTTWAA, & HHHTWAAA, & HHHWAAAA, & HPVAAAAA, \\
IIIVAAAA, & IIIYVAAA, & IIIYYVAA, & IIIYYYVA, & IIIYYYYV, & IIIYYYYY, \\
KKKAAAAA, & KLQAAAAA, & LLLAAAAA, & MKYAAAAA, & MMMMAAAA, & NNNCCCCC, \\
NNNCCCCQ, & NNNCCCQA, & NNNCCQAA, & NNNCQAAA, & NNNQAAAA, & NSTAAAAA, \\
PPPAAAAA, & QPKAAAAA, & QQQKAAAA, & QQQLKAAA, & QQQLLKAA, & QQQLLLKA, \\
QQQLLLLK, & QQQLLLLL, & QYEAAAAA, & RRRDAAAA, & RRRNDAAA, & RRRNNDAA, \\
RRRNNNDA, & RRRNNNND, & RRRNNNNN, & SISAAAAA, & SVTAAAAA, & TTCAAAAA, \\
VFRAAAAA, & WMPAAAAA, & WWDAAAAA, & YGLAAAAA & &
\end{array}\right\}$$

- For 25.6 billion vertices in $H_{8,20}$, around 600 quintillion pairwise checks per node would be required.

One-Hot Encodings

Definition (One-Hot encoding)

The **one-hot encoding** of a vertex v in $H_{k,a}$, is an $a \times k$ binary matrix V where $V_{j,i} = 1$ if $j = v[i]$, otherwise $V_{j,i} = 0$.

One-Hot Encodings

Definition (One-Hot encoding)

The **one-hot encoding** of a vertex v in $H_{k,a}$, is an $a \times k$ binary matrix V where $V_{j,i} = 1$ if $j = v[i]$, otherwise $V_{j,i} = 0$.

Example:

The one-hot encoding of 1213 in $H_{4,3}$ is the matrix $\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$.

Computing Hamming distance with One-Hot encodings

Theorem (Laird, Tillquist, Lladser)

For any two vertices a and b in $H_{k,a}$, if A and B denote their one-hot encodings, respectively, then:

$$d(a, b) = k - Tr(A^T B)$$

Computing Hamming distance with One-Hot encodings

Theorem (Laird, Tillquist, Lladser)

For any two vertices a and b in $H_{k,a}$, if A and B denote their one-hot encodings, respectively, then:

$$d(a, b) = k - Tr(A^T B)$$

Example: $v_1 = 1223$, $v_2 = 2123$

Computing Hamming distance with One-Hot encodings

Theorem (Laird, Tillquist, Lladser)

For any two vertices a and b in $H_{k,a}$, if A and B denote their one-hot encodings, respectively, then:

$$d(a,b) = k - Tr(A^T B)$$

Example: $v_1 = 1223$, $v_2 = 2123$

$$Tr(V_1^T V_2) = Tr(\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}) = Tr(\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}) = 2$$

Applying the Theorem to Resolvability

Let R be a subset of vertices in $H_{k,a}$, and $V_1, ..., V_n$ be the column vectorized one-hot encodings of the vertices in R. Define:

$$A := \begin{pmatrix} V_1^T \\ \vdots \\ V_n^T \end{pmatrix}$$

Applying the Theorem to Resolvability

Let R be a subset of vertices in $H_{k,a}$, and $V_1, ..., V_n$ be the column vectorized one-hot encodings of the vertices in R. Define:

$$A := \begin{pmatrix} V_1^T \\ \vdots \\ V_n^T \end{pmatrix}$$

For $x, y \in H_{k,a}$ with X, Y column vectorized one-hot encodings:

$$A(X - Y) = \begin{bmatrix} d(r_1, y) - d(r_1, x) \\ \vdots \\ d(r_n, y) - d(r_n, x) \end{bmatrix}$$

Therefore, x, y are resolved by R iff $A(X - Y) \neq 0$.

Main Result

Theorem (Laird, Tillquist, Lladser)

Let $R = \{r_1, \ldots, r_n\}$ be a subset of vertices in $H_{k,a}$ of cardinality n. If for each $1 \leq i \leq n$, V_i denotes the column vectorized one-hot encoding of r_i, and we define the matrix

$$A := \begin{pmatrix} V_1^T \\ \vdots \\ V_n^T \end{pmatrix}$$

then R **is resolving iff there exists no non-trivial solution z to the system $Az = 0$ satisfying the following constraints:** *if z is decomposed into k blocks of length a as follows:*

$$\left((z_1, \ldots, z_a), (z_{a+1}, \ldots, z_{2a}), \ldots, (z_{(k-1)a+1}, \ldots, z_{ka})\right)^T,$$

then each block is identically zero, or it has exactly one 1 and one -1 entry and all other entries are 0.

Example of Theorem

Consider the resolving set $R = \{100, 101, 001\}$ on $H_{3,2}$.

Example of Theorem

Consider the resolving set $R = \{100, 101, 001\}$ on $H_{3,2}$.

$$\mathrm{A} = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}, \text{ so: } \operatorname{rref}(\mathrm{A}) = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

Example of Theorem

Consider the resolving set $R = \{100, 101, 001\}$ on $H_{3,2}$.

$$\mathrm{A} = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}, \text{ so: } \operatorname{rref}(\mathrm{A}) = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

With $z = \big((z_1, z_2), (z_3, z_4), (z_5, z_6)\big)$, $\operatorname{rref}(\mathrm{A})$ gives:

$$z_1 = -z_3 - z_6;\ z_2 = -z_3 - z_6;\ z_5 = z_6.$$

Example of Theorem

Consider the resolving set $R = \{100, 101, 001\}$ on $H_{3,2}$.

$$\mathrm{A} = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}, \text{ so: } \mathrm{rref(A)} = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

With $z = \big((z_1, z_2), (z_3, z_4), (z_5, z_6)\big)$, rref(A) gives:

$$z_1 = -z_3 - z_6;\ z_2 = -z_3 - z_6;\ z_5 = z_6.$$

Going through constraints on z_3, z_4, z_6 will give the values for z_1, z_2, z_5.

1. $z_5 = z_6$ implies $z_5 = z_6 = 0$ since z_5, z_6 are in the same block.
2. $z_1 = z_2 = -z_3$ so $z_1 = z_2 = 0$ since z_1, z_2 are in the same block.
3. $z_3 = 0$ implies $z_4 = 0$ since z_3, z_4 are in the same block.

Therefore only the trivial solution $z = 0$ exists and R is resolving on $H_{3,2}$.

Automatically Handling Constraints

$$P = \left\{ \begin{array}{rl} z_1(z_1-1)(z_1+1) & = 0 \\ & \vdots \\ z_{ak}(z_{ak}-1)(z_{ak}+1) & = 0 \end{array} \right.$$

Automatically Handling Constraints

$$P = \begin{cases} z_1(z_1-1)(z_1+1) = 0 \\ \vdots \\ z_{ak}(z_{ak}-1)(z_{ak}+1) = 0 \\ \hline \sum_{i=1}^{a} z_i = 0 \\ \vdots \\ \sum_{(k-1)a+1}^{ka} z_i = 0 \\ \hline \end{cases}$$

Automatically Handling Constraints

$$P = \begin{cases} z_1(z_1-1)(z_1+1) = 0 \\ \vdots \\ z_{ak}(z_{ak}-1)(z_{ak}+1) = 0 \\ \hline \sum_{i=1}^{a} z_i = 0 \\ \vdots \\ \sum_{(k-1)a+1}^{ka} z_i = 0 \\ \hline (\sum_{i=1}^{a} z_i^2)(2 - \sum_{i=1}^{a} z_i^2) = 0 \\ \vdots \\ (\sum_{i=(k-1)a+1}^{ka} z_i^2)(2 - \sum_{i=(k-1)a+1}^{ka} z_i^2) = 0 \end{cases}$$

Automatically Handling Constraints

$$P = \left\{ \begin{array}{rl} z_1(z_1-1)(z_1+1) & = 0 \\ \vdots & \\ z_{ak}(z_{ak}-1)(z_{ak}+1) & = 0 \\ \hline \sum_{i=1}^{a} z_i & = 0 \\ \vdots & \\ \sum_{(k-1)a+1}^{ka} z_i & = 0 \\ \hline (\sum_{i=1}^{a} z_i^2)(2-\sum_{i=1}^{a} z_i^2) & = 0 \\ \vdots & \\ (\sum_{i=(k-1)a+1}^{ka} z_i^2)(2-\sum_{i=(k-1)a+1}^{ka} z_i^2) & = 0 \end{array} \right.$$

Resolvability is equivalent to showing $\{P = 0\} \cap ker(A)$ is only the trivial solution $z = 0$.

Solving the Polynomial System

Definition ([4, 5])

For a set of polynomials in the variable $z = (z_1, \ldots, z_k)$, $P = \{p_1(z), \ldots, p_n(z)\}$, the associated **polynomial ideal**, denoted $I(P)$, is defined as the set of all polynomials of the form $f_1 p_1 + \ldots + f_n p_n$, where $f_i(z)$ are arbitrary polynomials.

Solving the Polynomial System

Definition ([4, 5])

For a set of polynomials in the variable $z = (z_1, \ldots, z_k)$, $P = \{p_1(z), \ldots, p_n(z)\}$, the associated **polynomial ideal**, denoted $I(P)$, is defined as the set of all polynomials of the form $f_1 p_1 + \ldots + f_n p_n$, where $f_i(z)$ are arbitrary polynomials.

- z is a solution of $P = 0$ iff, for each $g \in I(P)$, $g(z) = 0$.
- It is often more convenient to characterize the roots of $I(P)$ instead of P itself.

Solving the Polynomial System

Definition ([4, 5])

For a set of polynomials in the variable $z = (z_1, \ldots, z_k)$, $P = \{p_1(z), \ldots, p_n(z)\}$, the associated **polynomial ideal**, denoted $I(P)$, is defined as the set of all polynomials of the form $f_1 p_1 + \ldots + f_n p_n$, where $f_i(z)$ are arbitrary polynomials.

- z is a solution of $P = 0$ iff, for each $g \in I(P)$, $g(z) = 0$.
- It is often more convenient to characterize the roots of $I(P)$ instead of P itself.

Gröbner bases can be intuitively understood as orthonormal bases for polynomial ideals of the form $I(P)$.

Important Definitions for Gröbner Bases

Definition ([4, 5])

A **monomial** in z is defined as any product of the form $z_1^{a_1} \cdots z_k^{a_k}$, with non-negative integers $a_1, \ldots, a_k$. This product is often written as z^a, with $a = (a_1, \ldots, a_k)$.

Important Definitions for Gröbner Bases

Definition ([4, 5])

A **monomial** in z is defined as any product of the form $z_1^{a_1} \cdots z_k^{a_k}$, with non-negative integers $a_1, \ldots, a_k$. This product is often written as z^a, with $a = (a_1, \ldots, a_k)$.

For what follows, we must define an ordering on monomials.
Lexicographic Ordering: $z^a > z^b$ if

- There is an index i such that $a_i > b_i$ and $a_j = b_j$ for all $j < i$
  Example:

$$z_1 z_4^2 > z_2 z_3^2$$
$$z_2^3 z_4 > z_2 z_4^3$$

Important Definitions for Gröbner Bases

Order the terms of a polynomial p in descending monomial order, $p = \alpha z^a + \beta z^b + \ldots$; in particular, $z^a > z^b$.

Important Definitions for Gröbner Bases

Order the terms of a polynomial p in descending monomial order, $p = \alpha z^a + \beta z^b + \ldots$; in particular, $z^a > z^b$.

- Leading Term: $LT(p) := \alpha z^a$
- Leading Monomial: $LM(p) := z^a$
- Leading Coefficient: $LC(p) := \alpha$

Important Definitions for Gröbner Bases

Order the terms of a polynomial p in descending monomial order, $p = \alpha z^a + \beta z^b + \ldots$; in particular, $z^a > z^b$.

- Leading Term: $LT(p) := \alpha z^a$
- Leading Monomial: $LM(p) := z^a$
- Leading Coefficient: $LC(p) := \alpha$

Definition ([4, 5])

The **S-Polynomial** of two polynomials p_1, p_2, denoted $spoly(p_1, p_2)$, is

$$spoly(p_1, p_2) := \frac{LCM(LM(p_1), LM(p_2))}{LT(p_1)} p_1 - \frac{LCM(LM(p_1), LM(p_2))}{LT(p_2)} p_2$$

Polynomial Reduction

Definition ([4, 5])

For a set of polynomials $P = \{p_1, \ldots, p_n\}$ and a monomial ordering $>$, any polynomial f can be written:

$$f = q_1 p_1 + \ldots + q_n p_n + r,$$

where r is such that no monomial in r is divisible by any $LM(p_i)$.

This is called **reducing** f by P, denoted $f \xrightarrow{P} r$. q_i are called **quotients** and r is called the **remainder** or **reduction**.

Polynomial Reduction

Definition ([4, 5])

For a set of polynomials $P = \{p_1, \ldots, p_n\}$ and a monomial ordering $>$, any polynomial f can be written:

$$f = q_1 p_1 + \ldots + q_n p_n + r,$$

where r is such that no monomial in r is divisible by any $LM(p_i)$.
This is called **reducing** f by P, denoted $f \xrightarrow{P} r$. q_i are called **quotients** and r is called the **remainder** or **reduction**.

The reduction $f \xrightarrow{P} r$ is not unique in general since the order in which the polynomials in P are processed matters.

Buchberger's Criterion

Definition

(Buchberger's Criterion.) A set of polynomials $G = \{g_1, \ldots, g_n\}$ is a **Gröbner basis** for a polynomial ideal I if and only if $I(G) = I$ and for all pairs $g_i, g_j \in G$:

$$Spoly(g_i, g_j) \xrightarrow{G} 0.$$

A Gröbner basis G is called **reduced** if for all distinct $g_i, g_j \in G$, $LC(g_i) = 1$, and no monomial of g_j is divisible by $LT(g_i)$.

Buchberger's Criterion

Definition

(Buchberger's Criterion.) A set of polynomials $G = \{g_1, \ldots, g_n\}$ is a **Gröbner basis** for a polynomial ideal I if and only if $I(G) = I$ and for all pairs $g_i, g_j \in G$:

$$Spoly(g_i, g_j) \xrightarrow{G} 0.$$

A Gröbner basis G is called **reduced** if for all distinct $g_i, g_j \in G$, $LC(g_i) = 1$, and no monomial of g_j is divisible by $LT(g_i)$.

- Under a given monomial ordering, the reduced Gröbner basis is unique.
- For a Gröbner basis G and polynomial f, $f \xrightarrow{G} 0$ iff $f \in I(G)$.

What were we doing again?

$$A = \begin{pmatrix} V_1^T \\ \vdots \\ V_n^T \end{pmatrix} \qquad Az = 0$$

What were we doing again?

$$A = \begin{pmatrix} V_1^T \\ \vdots \\ V_n^T \end{pmatrix} \qquad Az = 0$$

$$P = \begin{cases} z_1(z_1-1)(z_1+1) = 0 \\ \vdots \\ z_{ak}(z_{ak}-1)(z_{ak}+1) = 0 \\ \hline \sum_{i=1}^{a} z_i = 0 \\ \vdots \\ \sum_{(k-1)a+1}^{ka} z_i = 0 \\ \hline (\sum_{i=1}^{a} z_i^2)(2 - \sum_{i=1}^{a} z_i^2) = 0 \\ \vdots \\ (\sum_{i=(k-1)a+1}^{ka} z_i^2)(2 - \sum_{i=(k-1)a+1}^{ka} z_i^2) = 0 \end{cases}$$

Hilbert's Weak Nullstellensatz

Theorem (Hilbert's Weak Nullstellensatz [4, 6])

Let $P = \{p_1, ..., p_k\}$ be a set of polynomials with reduced Gröbner basis G. The solution set $\{P = 0\} = \emptyset$ if and only if $G = \{1\}$.

Hilbert's Weak Nullstellensatz

Theorem (Hilbert's Weak Nullstellensatz [4, 6])

Let $P = \{p_1, ..., p_k\}$ be a set of polynomials with reduced Gröbner basis G. The solution set $\{P = 0\} = \emptyset$ if and only if $G = \{1\}$.

- We need to check if the system $P \cup Az = 0$ has any non-trivial solutions to show whether R is resolving on $H_{k,a}$.

Hilbert's Weak Nullstellensatz

Theorem (Hilbert's Weak Nullstellensatz [4, 6])

Let $P = \{p_1, ..., p_k\}$ be a set of polynomials with reduced Gröbner basis G. The solution set $\{P = 0\} = \emptyset$ if and only if $G = \{1\}$.

- We need to check if the system $P \cup Az = 0$ has any non-trivial solutions to show whether R is resolving on $H_{k,a}$.
- Define $f_i = \sum_{j=1}^{ak} z_j^2 - 2i = 0$, and $P_i = P \cup f_i$ for $i = 1, \ldots, k$. $P_i \cup Az = 0$ will have only non-trivial solutions.

Hilbert's Weak Nullstellensatz

Theorem (Hilbert's Weak Nullstellensatz [4, 6])

Let $P = \{p_1, ..., p_k\}$ be a set of polynomials with reduced Gröbner basis G. The solution set $\{P = 0\} = \emptyset$ if and only if $G = \{1\}$.

- We need to check if the system $P \cup Az = 0$ has any non-trivial solutions to show whether R is resolving on $H_{k,a}$.
- Define $f_i = \sum_{j=1}^{ak} z_j^2 - 2i = 0$, and $P_i = P \cup f_i$ for $i = 1, \ldots, k$. $P_i \cup Az = 0$ will have only non-trivial solutions.
- Let G_i be the reduced Gröbner basis of $P_i \cup Az$. R is resolving iff $G_i = \{1\}$ for all $i = 1, \ldots, k$.

Leveraging the Structure of P

We can partition P into disjoint sets (which we call blocks) as follows. For each $i = 0, \ldots, (k-1)$, define:

$$P_i := \left\{ \begin{array}{c} z_{ia+1}(z_{ia+1}-1)(z_{ia+1}+1) \\ \vdots \\ z_{(i+1)a}(z_{(i+1)a}-1)(z_{(i+1)a}+1) \\ \frac{\sum_{j=ia+1}^{(i+1)a} z_j}{(\sum_{j=ia+1}^{(i+1)a} z_j^2)(2-\sum_{j=ia+1}^{(i+1)a} z_j^2)} \end{array} \right.$$

Then $P = P_0 \cup \ldots \cup P_{k-1}$.
Let z_i be the variables for the block P_i; in particular, $z_i \cap z_j = \emptyset$ for $i \neq j$.

Leveraging the Structure of P

Since the variables of different blocks are disjoint, we can use the following:

Lemma ([5])

Consider two polynomial sets P_1 and P_2 with disjoint variables x and y, and reduced Gröbner bases $G_1 \neq \{1\}$ and $G_2 \neq \{1\}$, respectively. Then $(G_1 \cup G_2)$ is a reduced Gröbner basis of $(P_1 \cup P_2)$.
This extends via induction to any finite unions of polynomial sets that do not share variables.

Leveraging the Structure of P

Since the variables of different blocks are disjoint, we can use the following:

Lemma ([5])

Consider two polynomial sets P_1 and P_2 with disjoint variables x and y, and reduced Gröbner bases $G_1 \neq \{1\}$ and $G_2 \neq \{1\}$, respectively. Then $(G_1 \cup G_2)$ is a reduced Gröbner basis of $(P_1 \cup P_2)$.
This extends via induction to any finite unions of polynomial sets that do not share variables.

Applying this lemma to $P = P_0 \cup \ldots \cup P_{k-1}$ gives:

Corollary (Laird, Tillquist, Lladser)

Let G_i be the reduced Gröbner basis of P_i. Then $G = (G_0 \cup \ldots \cup G_{k-1})$ is the reduced Gröbner basis of P.

Leveraging the Structure of P

Every block P_i is equivalent to block P_0 under the variable change:

$$z_{ia+j} \longrightarrow z_j$$

Leveraging the Structure of P

Every block P_i is equivalent to block P_0 under the variable change:

$$z_{ia+j} \longrightarrow z_j$$

$$\begin{cases} z_{ia+1}(z_{ia+1}-1)(z_{ia+1}+1) \\ \vdots \\ \dfrac{z_{(i+1)a}(z_{(i+1)a}-1)(z_{(i+1)a}+1)}{\sum_{j=ia+1}^{(i+1)a} z_j} \\ \dfrac{}{(\sum_{j=ia+1}^{(i+1)a} z_j^2)(2-\sum_{j=ia+1}^{(i+1)a} z_j^2)} \end{cases} \longrightarrow \begin{cases} z_1(z_1-1)(z_1+1) \\ \vdots \\ \dfrac{z_a(z_a-1)(z_a+1)}{\sum_{j=1}^{a} z_j} \\ \dfrac{}{(\sum_{j=1}^{a} z_j^2)(2-\sum_{j=1}^{a} z_j^2)} \end{cases}$$

Leveraging the Structure of P

Every block P_i is equivalent to block P_0 under the variable change:

$$z_{ia+j} \longrightarrow z_j$$

$$\begin{cases} z_{ia+1}(z_{ia+1}-1)(z_{ia+1}+1) \\ \vdots \\ \dfrac{z_{(i+1)a}(z_{(i+1)a}-1)(z_{(i+1)a}+1)}{\sum_{j=ia+1}^{(i+1)a} z_j} \\ \dfrac{}{(\sum_{j=ia+1}^{(i+1)a} z_j^2)(2-\sum_{j=ia+1}^{(i+1)a} z_j^2)} \end{cases} \longrightarrow \begin{cases} z_1(z_1-1)(z_1+1) \\ \vdots \\ \dfrac{z_a(z_a-1)(z_a+1)}{\sum_{j=1}^{a} z_j} \\ \dfrac{}{(\sum_{j=1}^{a} z_j^2)(2-\sum_{j=1}^{a} z_j^2)} \end{cases}$$

So it is only necessary to compute G_0 since reversing the variable change on G_0 yields G_i.

Explicit Pattern Representation of G_0

P_0 only changes with the alphabet size a, and increasing k simply adds another block to P. Studying how G_0 changes with a will give insight into all of G.

Explicit Pattern Representation of G_0

P_0 only changes with the alphabet size a, and increasing k simply adds another block to P. Studying how G_0 changes with a will give insight into all of G.

- Used Python computer algebra package SymPy [7] to compute Gröbner bases.
- Studying the computed Gröbner bases for small values of a illuminated an explicit pattern.

Explicit Pattern Representation of G_0

P_0 only changes with the alphabet size a, and increasing k simply adds another block to P. Studying how G_0 changes with a will give insight into all of G.

- Used Python computer algebra package SymPy [7] to compute Gröbner bases.
- Studying the computed Gröbner bases for small values of a illuminated an explicit pattern.

$$G_0 = \begin{cases} \sum_{i=1}^{a} z_i \\ z_2(z_2-1)(z_2+1) \\ \vdots \\ \dfrac{z_a(z_a-1)(z_a+1)}{z_2 z_3(z_2+z_3)} \\ \vdots \\ \dfrac{z_{a-1}z_a(z_{a-1}+z_a)}{z_2 z_3 z_4} \\ \vdots \\ z_{a-2}z_{a-1}z_a \end{cases}$$

Simplifying the System for Hypercubes

The general approach on $H_{k,a}$ can be simplified for hypercubes, $H_{k,2}$.

Simplifying the System for Hypercubes

The general approach on $H_{k,a}$ can be simplified for hypercubes, $H_{k,2}$.

Consider the first block of a polynomial system for $H_{k,2}$:

$$P_0 = \begin{cases} z_1(z_1-1)(z_1+1) = 0 \\ z_2(z_2-1)(z_2+1) = 0 \\ \hline z_1 + z_2 = 0 \\ \hline (z_1^2+z_2^2)(2-(z_1^2+z_2^2)) = 0 \end{cases}$$

- Because of the block structure of P, any simplifications we perform on P_0 will be valid for every other block.

Simplifying the System for Hypercubes

$$P_0 = \begin{cases} z_1(z_1-1)(z_1+1) = 0 \\ z_2(z_2-1)(z_2+1) = 0 \\ \hline z_1 + z_2 = 0 \\ \hline (z_1^2+z_2^2)(2-(z_1^2+z_2^2)) = 0 \end{cases}$$

Simplifying the System for Hypercubes

$$P_0 = \begin{cases} z_1(z_1-1)(z_1+1) &= 0 \\ z_2(z_2-1)(z_2+1) &= 0 \\ \hline z_1+z_2 &= 0 \\ \hline (z_1^2+z_2^2)(2-(z_1^2+z_2^2)) &= 0 \end{cases}$$

The third polynomial gives $z_1 = -z_2$ and $z_1^2 = z_2^2$.

Substitution into the first and fourth polynomials gives:

$$z_1(z_1-1)(z_1+1) = -(z_2^3 - z_2)$$
$$(z_1^2+z_2^2)(2-(z_1^2+z_2^2)) = -4z_2(z_2^3 - z_2)$$

Simplifying the System for Hypercubes

$$P_0 = \begin{cases} z_1(z_1-1)(z_1+1) &= 0 \\ z_2(z_2-1)(z_2+1) &= 0 \\ \hline z_1+z_2 &= 0 \\ \hline (z_1^2+z_2^2)(2-(z_1^2+z_2^2)) &= 0 \end{cases}$$

The third polynomial gives $z_1 = -z_2$ and $z_1^2 = z_2^2$.

Substitution into the first and fourth polynomials gives:

$$z_1(z_1-1)(z_1+1) = -(z_2^3 - z_2)$$
$$(z_1^2+z_2^2)(2-(z_1^2+z_2^2)) = -4z_2(z_2^3-z_2)$$

These redundancies lead to a simplified but equivalent polynomial system:

$$P_0 = \begin{cases} z_2(z_2-1)(z_2+1) &= 0 \\ \hline z_1+z_2 &= 0 \end{cases}$$

Simplifying the System for Hypercubes

We can apply the same simplification to every block of P. Additionally, since $z_{i-1} + z_i = 0$ are linear equations, they can be removed from the polynomial system and transferred to the linear system.

Simplifying the System for Hypercubes

We can apply the same simplification to every block of P. Additionally, since $z_{i-1} + z_i = 0$ are linear equations, they can be removed from the polynomial system and transferred to the linear system.

Recall the matrix A from our main result, and let A_i be its i-th column:

$$Az = \begin{pmatrix} | & | & & | & | \\ A_1 & A_2 & \dots & A_{2k-1} & A_{2k} \\ | & | & & | & | \end{pmatrix} \begin{bmatrix} z_1 \\ \vdots \\ z_{2k} \end{bmatrix} = 0$$

Applying the linear functions $z_{i-1} + z_i = 0$ results in a simplified system:

$$\begin{pmatrix} | & & | \\ (A_2 - A_1) & \dots & (A_{2k} - A_{2k-1}) \\ | & & | \end{pmatrix} \begin{bmatrix} z_2 \\ \vdots \\ z_{2k} \end{bmatrix} = 0$$

Simplifying the System for Hypercubes

The simplifications, after renaming variables $z_{2i} \longrightarrow z_i$, give the following:

Corollary (Laird, Tillquist, Lladser)

Let $R = \{v_1, \ldots, v_n\}$ be a set of nodes in $H_{k,2}$, and $\bar{v}_i$ denote the logical negation (flip) of v_i. Consider the matrix of dimensions $(n \times k)$ defined as:

$$B := \begin{pmatrix} v_1 - \bar{v}_1 \\ \vdots \\ v_n - \bar{v}_n \end{pmatrix}.$$

Then, R resolves $H_{k,2}$ iff the equation $Bz = 0$, with $z \in \{-1, 0, 1\}^k$, has only a trivial solution.

- The corollary reproduces the system developed by A. F. Beardon [1]

Preliminary Runtime Analysis

We created 4,359 example sets on small Hamming graphs to test the theory. Approximately 200 sets were generated for $k = 1, \ldots, 10$ and $a = 2, \ldots, 5$, with $ak \leq 25$ to limit overall complexity.

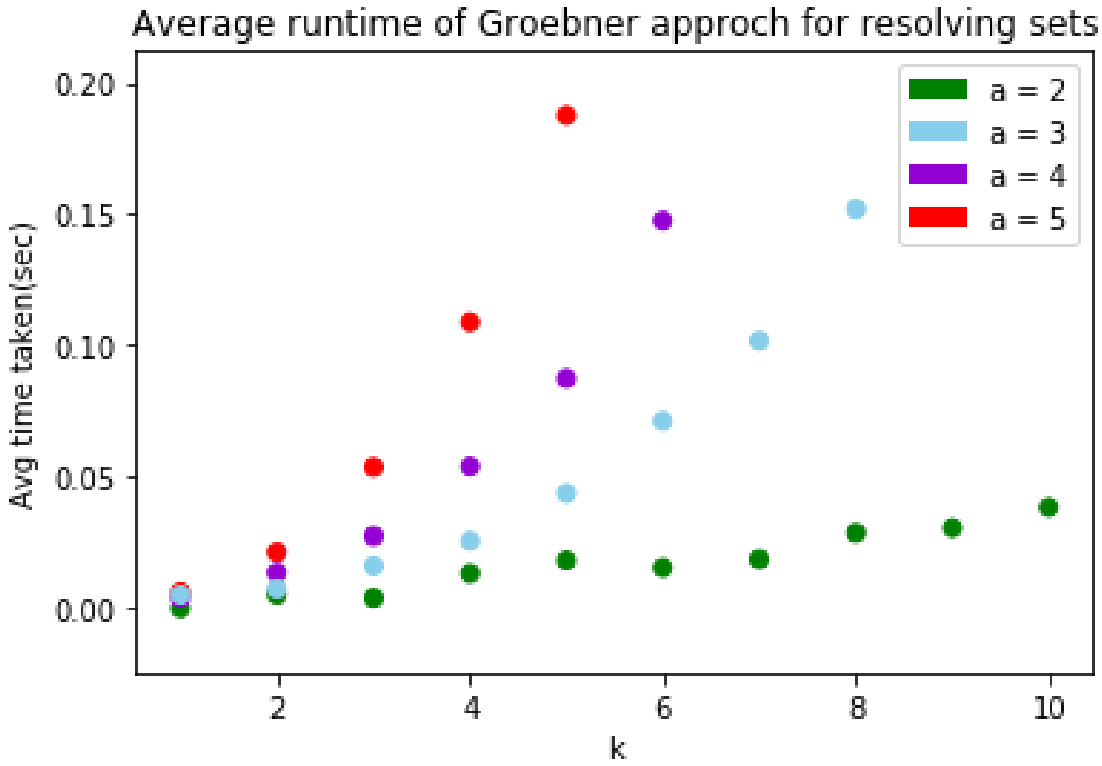

Figure: Runtime for Gröbner basis approach on resolving sets

Runtime Comparison

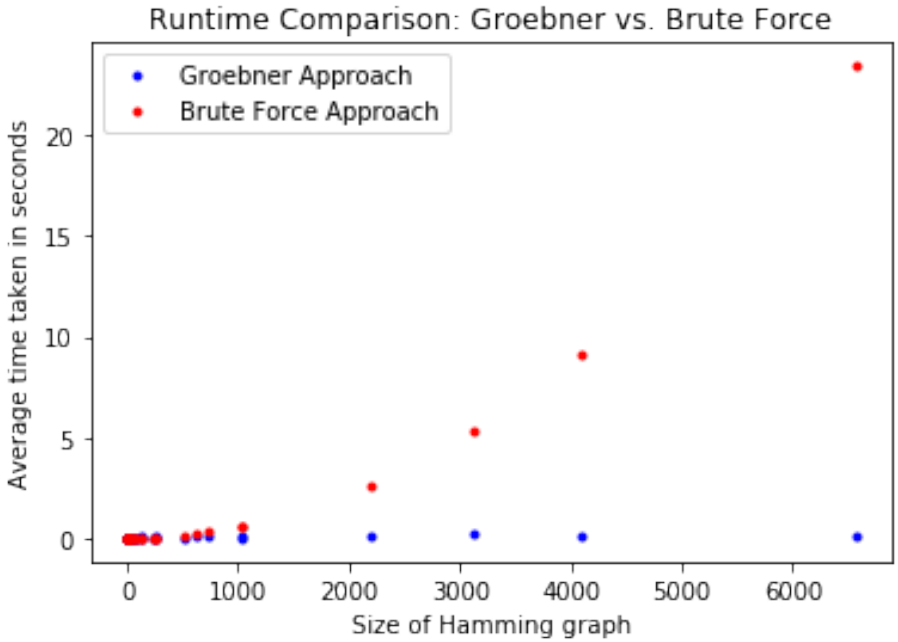

Figure: Runtime comparison on only resolving sets

Runtime Comparison

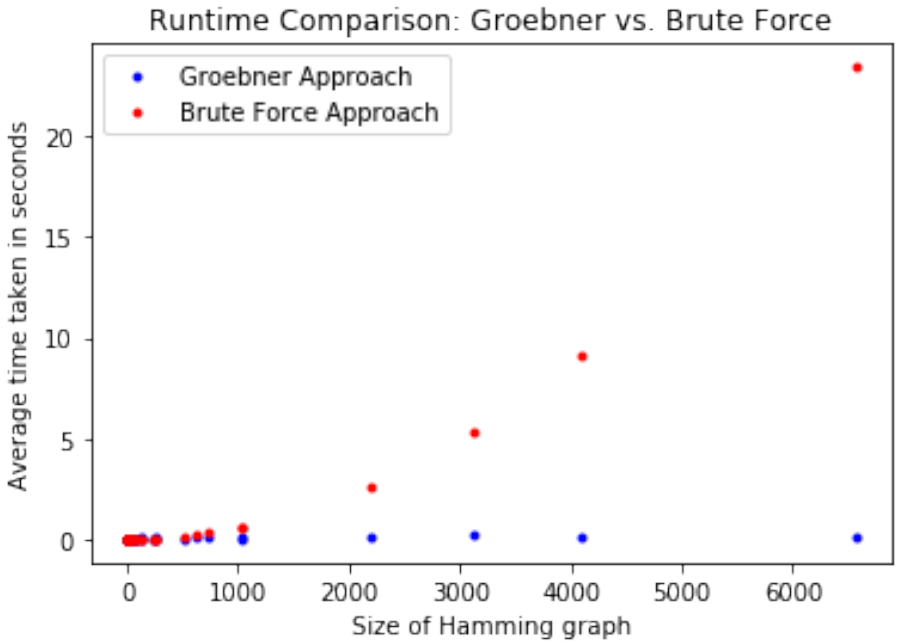

Figure: Runtime comparison on only resolving sets

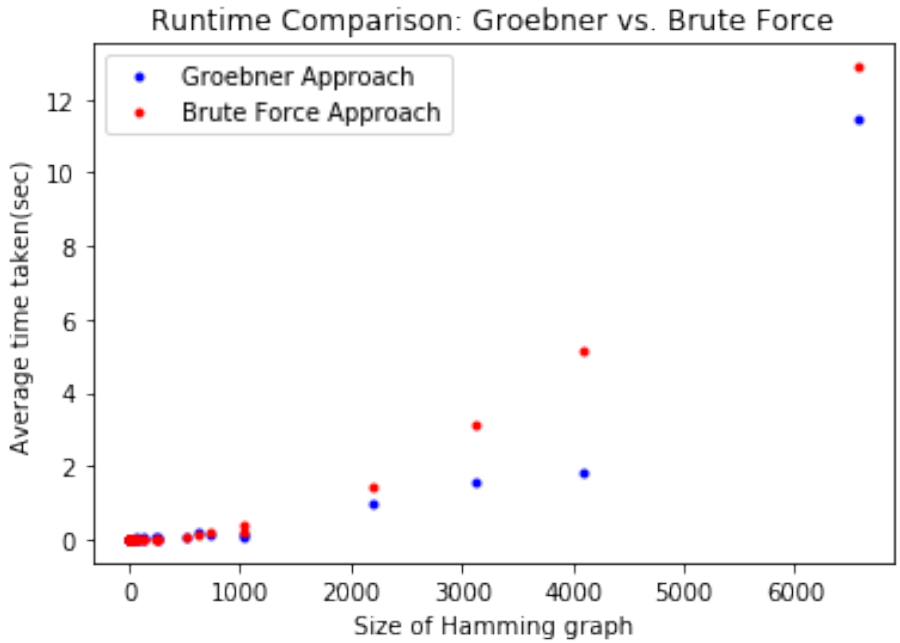

Figure: Runtime comparison on resolving and non-resolving sets

Runtime Comparison

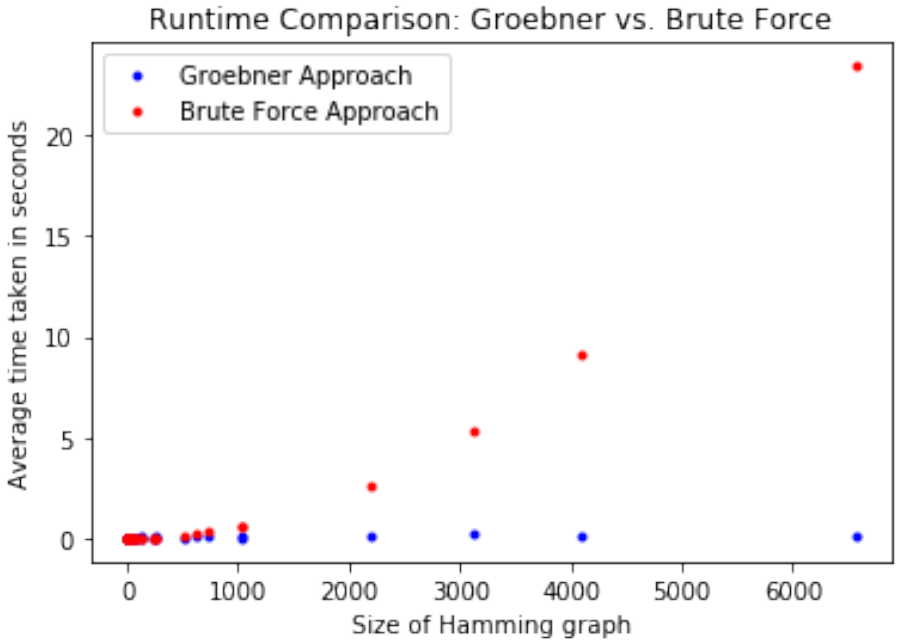

Figure: Runtime comparison on only resolving sets

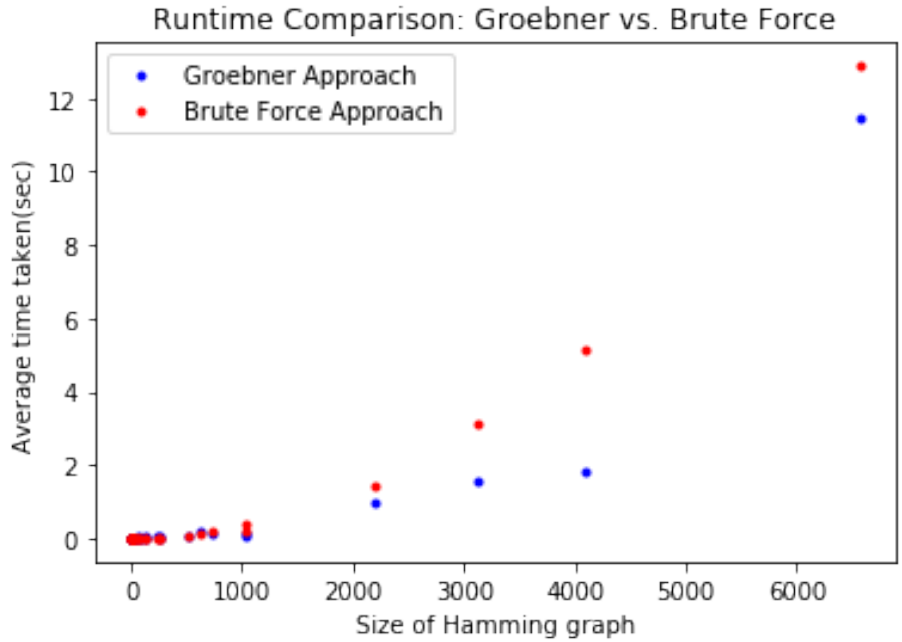

Figure: Runtime comparison on resolving and non-resolving sets

- Gröbner basis algorithm performance on non-resolving sets is on average longer than on resolving sets.
- Slow performance could be caused by the sets' associated linear systems or due to known, unresolved issues with SymPy.

Current and Future Work

- Study the block structure of the polynomial system to further optimize Gröbner basis computations.
- Study structure of the Gröbner bases G_i and try to derive a linear system which reduces them to $\{1\}$.

Current and Future Work

- Study the block structure of the polynomial system to further optimize Gröbner basis computations.
- Study structure of the Gröbner bases G_i and try to derive a linear system which reduces them to $\{1\}$.
- Explore if simplifications similar to the hypercubes exist on other alphabet sizes.

Current and Future Work

- Study the block structure of the polynomial system to further optimize Gröbner basis computations.
- Study structure of the Gröbner bases G_i and try to derive a linear system which reduces them to $\{1\}$.
- Explore if simplifications similar to the hypercubes exist on other alphabet sizes.
- Implement algorithms on $H_{8,20}$.
- Further computational complexity analysis.

Current and Future Work

- Study the block structure of the polynomial system to further optimize Gröbner basis computations.
- Study structure of the Gröbner bases G_i and try to derive a linear system which reduces them to $\{1\}$.
- Explore if simplifications similar to the hypercubes exist on other alphabet sizes.
- Implement algorithms on $H_{8,20}$.
- Further computational complexity analysis.
- Check how reducing the size of resolving sets impacts quality of embeddings for machine learning algorithms.

Acknowledgements

- Professor Manuel Lladser (Advisor)
- Richard Carter Tilquist (Co-Advisor)
- Stephen Becker and Rafael Frongillo (Committee Members)
- This research has been partially funded by NSF IIS grant 1836914

References I

[1] A. F. Beardon, "Resolving the Hypercube," *Discrete Applied Mathematics*, vol. 161, pp. 1882–1887, 2013.

[2] J. Cáceres, C. Hernando, M. Mora, I. M. Pelayo, M. L. Puertes, C. Seara, and D. R. Wood, "On the metric dimension of cartesian products of graphs," *Siam Journal of Discrete Math*, vol. 21, no. 2, pp. 423–441, 2007.

[3] R. C. Tillquist and M. E. Lladser, "Low-dimensional representation of genomic sequences," *Journal of Mathematical Biology*, pp. 1–29, 3 2019.

[4] D. Cox, J. Little, and D. O'Shea, *Using Algebraic Geometry*, vol. 1 of *Graduate Texts in Mathematics*.
Springer-Verlag New York, 1998.

[5] D. A. Cox, J. Little, and D. O'Shea, *Gröbner Bases*, pp. 49–119.
Cham: Springer International Publishing, 2015.

References II

[6] T. Tao, "Hilbert's nullstellensatz," November 2007.

[7] A. Meurer, C. P. Smith, M. Paprocki, O. Čertík, S. B. Kirpichev, M. Rocklin, A. Kumar, S. Ivanov, J. K. Moore, S. Singh, T. Rathnayake, S. Vig, B. E. Granger, R. P. Muller, F. Bonazzi, H. Gupta, S. Vats, F. Johansson, F. Pedregosa, M. J. Curry, A. R. Terrel, v. Roučka, A. Saboo, I. Fernando, S. Kulal, R. Cimrman, and A. Scopatz, "Sympy: symbolic computing in python," *PeerJ Computer Science*, vol. 3, p. e103, Jan. 2017.

[8] M. Hauptmann, R. Schmied, and C. Viehmann, "Approximation complexity of metric dimension problem," *Journal of Discrete Algorithms*, vol. 14, pp. 214–222, 2012.

[9] P. J. Slater, "Leaves of trees," *Congressus Numerantium*, vol. 14, pp. 549–559, 1975.

References III

[10] F. Haray and R. A. Melter, "On the metric dimension of a graph," *Ars Combinatoria*, vol. 2, pp. 191–195, 1976.

[11] V. Chvátal, "Mastermind," *Combinatorica*, vol. 3, no. 3, pp. 325–329, 1983.

[12] C. Hernando, M. Mora, I. M. Pelayo, and C. Seara, "Extremal graph theory for metric dimension and diameter," *Electronic Journal of Combinatorics*, vol. 17, no. R30, 2010.

[13] J. Cáceres, D. Garijo, M. L. Puertas, and C. Seara, "On the determining number and the metric dimension of graphs," *Electronic Journal of Combinatorics*, vol. 17, no. R63, 2010.

[14] J. Kratica, V. Kovačević-Vujčić, and M. Čangalović, "Computing the metric dimension of graphs by genetic algorithms," *Computational Optimization and Applications*, vol. 44, no. 2, p. 343, 2007.

References IV

[15] M. R. Garey and D. S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*.
W. H. Freeman & Co., 1990.

[16] F. A. Chaouche and A. Berrachedi, "Automorphisms group of generalized Hamming graphs," *Electronic Notes in Discrete Mathematics*, vol. 24, no. 9-15, 2006.

[17] P. Erdos and A. Rényi, "On two problems of information theory," *Magyar Tud. Akad. Mat. Kutató Int. Közl*, vol. 8, pp. 229–243, 1963.

[18] B. Lindström, "On a combinatory detection problem i," *I. Magyar Tud. Akad. Mat. Kutató Int. Közl*, vol. 9, pp. 195–207, 1964.

[19] J. C. Faugère, "A new efficient algorithm for computing Gröbner bases without reduction to zero (f5)," in *Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation*, ISSAC '02, (New York, NY, USA), pp. 75–83, ACM, 2002.

References V

[20] Y. Sun and D. Wang, "The F5 algorithm in Buchberger's style," *CoRR*, vol. abs/1006.5299, 2010.

[21] R. K. Guy and R. J. Nowakowski, "Coin-weighing problems," *The American Mathematical Monthly*, vol. 102, no. 2, pp. 164–167, 1995.

[22] L. M. Kelly and E. A. Nordhaus, "Distance sets in metric spaces," *Trans. Amer. Math. Soc.*, vol. 71, pp. 440–456, 1951.

[23] R. A. Horn, R. A. Horn, and C. R. Johnson, *Matrix analysis*. Cambridge university press, 1990.

[24] S. Khuller, B. Raghavachari, and A. Rosenfeld, "Landmarks in graphs," *Discrete Applied Mathematics*, vol. 70, pp. 217–229, 1996.

[25] J.-C. Faugère, "A new efficient algorithm for computing Gröbner bases F4," *Journal of Pure and Applied Algebra*, vol. 139, no. 1, pp. 61 – 88, 1999.

References VI

[26] C. Eder and J. E. Perry, "F5c: A variant of faugère's f5 algorithm with reduced gröbner bases," *J. Symb. Comput.*, vol. 45, no. 12, pp. 1442–1458, 2010.
MEGA'2009.

[27] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 855–864, ACM, 2016.

[28] W. J. Krzanowski, *Principles of Multivariate Analysis: A User's Perspective*.
OUP Oxford, 2000.

[29] P. J. Olver and C. Shakiban, *Linear Algebraic Systems*, pp. 1–74.
Cham: Springer International Publishing, 2018.