Gröbner-Gradient Decoding of Linear Codes

Edgar Martínez Moro
Applied Maths. Dept., Universidad de Valladolid
edgar@maf.uva.es
(Join work with M. Borges-Quintana, M. Borges-Trenard & P. Fitzpatrick)

Applicable Algebra and Computer Science:
A Meeting in Honor of Jacques Calmet
Universidad de la Rioja, February 28, 2008
Abstract

In this talk we will show a set of minimal codewords of binary linear code. This set can be computed by means of linear algebra from a Groebner basis of a binomial associated to the code (see [1] and the references therein). This set allows us to formulate a gradient-like complete decoding algorithm.

Main references:


Contents

Linear codes

Minimal subsets in codes

The Gröbner test set-(trial set)
   The Gröbner decoding algorithm

Worked example

Computing the Groebner basis
Linear codes

Let $\mathbb{F}_q$ be a finite field with $q$ elements and $\mathbb{F}_q^n$ be the $\mathbb{F}_q$-vector space of dimension $n$. We will call the vectors in $\mathbb{F}_q^n$ words.

A linear code $C$ of dimension $k$ and length $n$ is the image of a linear mapping $L : \mathbb{F}_q^k \to \mathbb{F}_q^n$, where $k \leq n$, i.e. $C = L(\mathbb{F}_q^k)$. The elements in $C$ are called codewords. There exists an $n \times (n - k)$ matrix $H$, called a parity check matrix, such that $c \cdot H = \mathbf{0}$ if and only if $c \in C$.

On the other hand, there exists a $k \times n$ generator matrix $G$ such that $G \cdot H = \mathbf{0}$.
The **Hamming weight** of a word \( y \) is the number of non-zero entries and we will denote it by \( \text{weight}(y) \). The Hamming distance between two words \( c_1, c_2 \) is \( d(c_1, c_2) = \text{weight}(c_1 - c_2) \) and the **minimum distance** \( d \) of a code is the minimum weight among all the non-zero codewords. The error correcting capability of a code is \( t = \left\lfloor \frac{d-1}{2} \right\rfloor \), where \( \lfloor \cdot \rfloor \) is the greatest integer function. Let \( B(C, t) = \{ y \in \mathbb{F}^n_2 \mid \exists c \in C \text{ s.t. } d(c, y) \leq t \} \), it is well-known that the equation
\[
e \cdot H = y \cdot H
\]
has a unique solution \( e \) with \( \text{weight}(e) \leq t \) for \( y \in B(C, t) \). The vector \( c = y - e \) is the codeword corresponding to \( y \) (the nearest codeword) and the vector \( e \) is called the **error vector**.
Minimal subsets in codes

Let $C \subseteq \mathbb{F}_q^n$ a linear code. We define the support of a codeword $c = (c_1, \ldots, c_n) \in C$ as

$$\text{supp}(c) = \{ i \in \{1, \ldots, n\} \mid c_i \neq 0 \} .$$  \hspace{1cm} (1)

If $\text{supp}(c') \subset \text{supp}(c)$ (respectively $\subseteq$) we will write $c' \prec c$ (respectively $\preceq$).
Minimal subsets in codes

Let \( C \subseteq \mathbb{F}_q^n \) a linear code. We define the **support of a codeword** \( \mathbf{c} = (c_1, \ldots, c_n) \in C \) as

\[
\text{supp}(\mathbf{c}) = \{ i \in \{1, \ldots, n\} \mid c_i \neq 0 \}. \tag{1}
\]

If \( \text{supp}(\mathbf{c}') \subset \text{supp}(\mathbf{c}) \) (respectively \( \subseteq \)) we will write \( \mathbf{c}' \prec \mathbf{c} \) (respectively \( \preceq \)).

**Definition**

A nonzero vector \( \mathbf{c} \in C \subseteq \mathbb{F}_q^n \) is said to be **minimal** if \( 0 \neq \mathbf{c}' \preceq \mathbf{c} \) and \( \mathbf{c}' \in C \) then it implies that there exist a nonzero constant \( \alpha \in \mathbb{F}_q \) such that \( \mathbf{c}' = \alpha \mathbf{c} \).
From now on we are in the **binary case** \((q = 2)\)
From now on we are in the **binary case** \((q = 2)\), i.e.

\[
\begin{align*}
c \in C & \text{ is } \textbf{minimal} \text{ if and only if there is no } c' \in C \setminus \{c\} \text{ such that } 0 \neq c' \preceq c.
\end{align*}
\]
The Gröbner test set-(trial set)

1. Complete decoding for a linear block code has proved to be a NP-hard computational problem.
The Gröbner test set-(trial set)

1. Complete decoding for a linear block code has proved to be a NP-hard computational problem.

2. Several attempts have been made to improve the syndrome decoding idea for a general linear code. They look for a smaller structure than the syndrome table to perform the decoding, they look for each coset the smaller weight of the words in that coset instead of storing the candidate error vector
   - Step-by-Step algorithm
   - Test set decoding (zero-neighbors and zero-guards).

We will call these procedures gradient-like decoding algorithms.
The Gröbner test set-(trial set)

1. Complete decoding for a linear block code has proved to be a NP-hard computational problem.

2. Several attempts have been made to improve the syndrome decoding idea for a general linear code. They look for a smaller structure than the syndrome table to perform the decoding, they look for each coset the smaller weight of the words in that coset instead of storing the candidate error vector
   - Step-by-Step algorithm
   - Test set decoding (zero-neighbors and zero-guards).

We will call these procedures gradient-like decoding algorithms.

3. In the same fashion we use the reduction given by a reduced Gröbner basis to give a procedure to decode any arbitrary binary linear code.
The Gröbner basis

Consider the polynomial ring $K[X]$, where $K$ is an arbitrary field and $[X]$ the set of terms in the variables $x_1, x_2, \cdots, x_n$. If $a = (a_1, \ldots, a_n) \in \mathbb{N}^n$ we will denote by $x^a$ the term $\prod_{i=1}^{n} x_i^{a_i}$. 
The Gröbner basis

Consider the polynomial ring $K[X]$, where $K$ is an arbitrary field and $[X]$ the set of terms in the variables $x_1, x_2, \cdots, x_n$. If $a = (a_1, \ldots, a_n) \in \mathbb{N}^n$ we will denote by $x^a$ the term $\prod_{i=1}^{n} x_i^{a_i}$.

Let $<$ be a fixed total degree compatible term order on $[X]$ with

$$x_1 < x_2 < \cdots < x_n.$$

$T(f)$ will denote the maximal term of a polynomial $f$ with respect to the order $<$ and $Td(f)$ the total degree of the maximal term $T(f)$ of $f$. If $I \subseteq K[X]$ is an ideal we denote by $T(I)$ the semigroup ideal in $[X]$ generated by $\{T(f) | f \in I\}$. Finally, if $F \subseteq K[X]$ then $\langle F \rangle$ denotes the polynomial ideal generated by $F$. 
Consider the morphism from \([X]\) onto \(\mathbb{F}_2^n\)

\[
\psi \left( x^\beta \right) = \psi \left( \prod_{i=1}^{n} x_i^{\beta_i} \right) = (\beta_1 \mod 2, \ldots, \beta_n \mod 2), \quad (2)
\]

\(\beta = (\beta_1, \ldots, \beta_n) \in \mathbb{N}^n\). We will use \(x_i\) to refer the indeterminate in the monoid or the associated vector \(e_i\) in \(\mathbb{F}_2^n\).
Consider the morphism from \([X]\) onto \(\mathbb{F}_2^n\)

\[
\psi\left(x^\beta\right) = \psi\left(\prod_{i=1}^{n} x_i^{\beta_i}\right) = (\beta_1 \mod 2, \ldots, \beta_n \mod 2),
\]

(2)

\(\beta = (\beta_1, \ldots, \beta_n) \in \mathbb{N}^n\). We will use \(x_i\) to refer the indeterminate in the monoid or the associated vector \(e_i\) in \(\mathbb{F}_2^n\).

Consider the equivalence relation \((x, y) \in R_C \iff x - y \in C\).

**Definition**

The binomial ideal \(I(C)\) associated with the code \(C\) is

\[
I(C) = \left\langle \{x^w - x^v \mid (\psi(x^w), \psi(x^u)) \in R_C\} \right\rangle \subseteq K[X].
\]

(3)
Let be \( \{w_1, \ldots, w_k\} \) be the row vectors of any matrix whose rows span the code \( C \) and

\[
I = \left\langle \{x^{w_1} - 1, \ldots, x^{w_k} - 1\} \cup \{x_i^2 - 1 \mid i = 1, \ldots, n\} \right\rangle \quad (4)
\]
Let be \( \{w_1, \ldots, w_k\} \) be the row vectors of any matrix whose rows span the code \( C \) and

\[
l = \left\langle \{x^{w_1} - 1, \ldots, x^{w_k} - 1\} \cup \{x_i^2 - 1 \mid i = 1, \ldots, n\} \right\rangle \tag{4}
\]

Since the set of vectors \( \{w_1, \ldots, w_k\} \) generates \( C \) as a \( \mathbb{F}_2 \)-vector space we have

\[
l = l(C).
\]
Let be \( \{w_1, \ldots, w_k\} \) be the row vectors of any matrix whose rows span the code \( C \) and

\[
I = \left\langle \{x^{w_1} - 1, \ldots, x^{w_k} - 1\} \cup \{x_i^2 - 1 \mid i = 1, \ldots, n\} \right\rangle \tag{4}
\]

Since the set of vectors \( \{w_1, \ldots, w_k\} \) generates \( C \) as a \( \mathbb{F}_2 \)-vector space we have

\[
I = I(C).
\]

Let \( G \) be the reduced Gröbner basis of the ideal \( I(C) \) with respect to the term ordering \( < \) and let \( g \in K[X] \), we denote by \( \text{Can}(g, G) \) the canonical form of \( g \) with respect to the Gröbner basis \( G \).
If \( g = x^w - x^v \in I(C) \) denote by \( c_g \) the codeword associated to the binomial \( g \), that is,

\[
c_g = \psi(x^w) + \psi(x^v)
\]
If \( g = x^w - x^v \in I(C) \) denote by \( c_g \) the codeword associated to the binomial \( g \), that is,

\[
c_g = \psi(x^w) + \psi(x^v)
\]

**Definition**

Let \( G \) be the reduced Gröbner basis with respect to the term ordering \( < \) of the binomial ideal \( I(C) \). The **Gröbner codewords set** is

\[
C_G = \{ c_g \mid g \in G \} \setminus \{0\}.
\]
Theorem

The elements of the set $C_G$ of Gröbner codewords are minimal codewords of the code $C$. 
Theorem

The elements of the set $C_G$ of Gröbner codewords are minimal code-words of the code $C$.

Proof.

Let $g = x^w - x^v \in G$ such that $x^w = T(g)$ and $x^v = \text{Can}(g, G)$, and $x^w$ is an irredundant generator of the semigroup ideal $T(I(C))$. 
Theorem

*The elements of the set $C_G$ of Gröbner codewords are minimal codewords of the code $C$.***

**Proof.**

Let $g = x^w - x^v \in G$ such that $x^w = T(g)$ and $x^v = \text{Can}(g, G)$, and $x^w$ is an irredundant generator of the semigroup ideal $T(I(C))$. Suppose $c_g \in C_G$ is not minimal, then there exists $c \in C$ such that $\text{supp}(c) \subset \text{supp}(c_g)$. 
Theorem

The elements of the set $C_G$ of Gröbner codewords are minimal codewords of the code $C$.

Proof.

Let $g = x^w - x^v \in G$ such that $x^w = T(g)$ and $x^v = \text{Can}(g, G)$, and $x^w$ is an irredudant generator of the semigroup ideal $T(I(C))$. Suppose $c_g \in C_G$ is not minimal, then there exists $c \in C$ such that $\text{supp}(c) \subset \text{supp}(c_g)$. Let $c_1 \in \mathbb{E}_2^n$ such that $\text{supp}(c_1) \subset \text{supp}(c)$ and $\text{supp}(c_1) \subset \text{supp}(\psi(x^w))$, thus $c_2 = c - c_1$ fulfill $\text{supp}(c_2) \subset \text{supp}(\psi(x^v))$. 
Theorem

The elements of the set \( C_G \) of Gröbner codewords are minimal codewords of the code \( C \).

Proof.

Let \( g = x^w - x^v \in G \) such that \( x^w = T(g) \) and \( x^v = \text{Can}(g, G) \), and \( x^w \) is an irredundant generator of the semigroup ideal \( T(I(C)) \).

Suppose \( c_g \in C_G \) is not minimal, then there exists \( c \in C \) such that \( \text{supp}(c) \subset \text{supp}(c_g) \). Let \( c_1 \in \mathbb{E}_n^2 \) such that \( \text{supp}(c_1) \subset \text{supp}(c) \) and \( \text{supp}(c_1) \subset \text{supp}(\psi(x^w)) \), thus \( c_2 = c - c_1 \) fullfils \( \text{supp}(c_2) \subset \text{supp}(\psi(x^v)) \).

Let \( x^u \) be the maximum between \( x^{c_1} \) and \( x^{c_2} \), thus \( x^u \in T(I(C)) \), \( \text{supp}(\psi(x^u)) \subset \text{supp}(c) \), and \( x^u \) divides \( x^w \) or \( x^v \) which is a contradiction.
The Gröbner decoding algorithm

The theorem allow us to perform a gradient-like decoding algorithm but according to $<$ instead of the weight of the vectors. Thus we say that the set of Gröbner codewords is a “test set”.

\begin{itemize}
  \item[i:] $i := 0$
  \item[v] $v_i := y$
  \item[c] $c_i := 0$
  \item[3.] Repeat
    \item[4.] Find $w \in \mathcal{C}_G$ such that $x v_i > x v_i + 1$ and $v_{i+1} = v_i + w$.
    \item[5.] $c_{i+1} = c_i + w$
    \item[i] $i := i + 1$
  \item[5.] Until such a $w$ does not exist.
  \item[6.] Return $[c_i]$.\end{itemize}
The Gröbner decoding algorithm

The theorem allow us to perform a gradient-like decoding algorithm but according to \(<\) instead of the weight of the vectors. Thus we say that the set of Gröbner codewords is a “test set”.

**Input:** $C_G$ and $y$ a received vector.

**Output:** One of the closest codewords to $y$.

1. $i := 0; v_i = y; c_i = 0$.
2. Repeat
3. Find $w \in C_G$ such that $x^{v_i} > x^{v_{i+1}}$ and $v_{i+1} = v_i + w$.
4. $c_{i+1} = c_i + w; i = i + 1$
5. Until such a $w$ does not exist.
6. Return[$c_i$].
Complexity

Preprocessing  Computing the reduced Gröbner basis or the border basis performed $O(n^22^{n-k})$ operations.
Complexity

Preprocessing  Computing the reduced Gröbner basis or the border basis performed $\mathcal{O}(n^22^{n-k})$ operations.

Decoding  The decoding complexity depends on the size of $C_G$ (or $C_B$) and the number of reductions. The number of reductions for $C_B$ is less than $n$. 
Complexity

Preprocessing  Computing the reduced Gröbner basis or the border basis performed $O(n^2 2^{n-k})$ operations.

Decoding  The decoding complexity depends on the size of $C_G$ (or $C_B$) and the number of reductions. The number of reductions for $C_B$ is less than $n$.

Computing $t$  The error correction capability of an arbitrary linear code (not necessarily binary) can be computed in at most $m \cdot n \cdot S(t + 1)$ iterations of the Algorithm showed in B., B. & M. where

$$S(l) = \sum_{i=0}^{l} \binom{n}{i} (q - 1)^i.$$
Consider the code $C$ in $\mathbb{F}_2^6$ with generator matrix:

$$G = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}.$$
Worked example

Consider the code $C$ in $\mathbb{F}_2^6$ with generator matrix:

$$G = \begin{pmatrix}
1 & 0 & 0 & 1 & 1 & 1 \\
0 & 1 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 1
\end{pmatrix}.$$

The set of codewords is

$$C = \{(0, 0, 0, 0, 0, 0), (1, 0, 1, 1, 0, 0), (1, 1, 0, 0, 1, 0), (0, 1, 0, 1, 0, 1), (0, 0, 1, 0, 1, 1), (1, 1, 1, 0, 0, 1), (0, 1, 1, 1, 1, 0), (1, 0, 0, 1, 1, 1)\}.$$
The reduced Gröbner basis of $I(C)$ with respect to the degree reverse Lexicographical ordering $<$ is
The reduced Gröbner basis of $I(C)$ with respect to the degree reverse Lexicographical ordering $<$ is

$$G = \{ x_1^2 - 1, x_2^2 - 1, x_3^2 - 1, x_4^2 - 1, x_5^2 - 1, x_6^2 - 1, $$
$$x_1x_2 - x_5, x_1x_3 - x_4, x_1x_4 - x_3, x_1x_5 - x_2, $$
$$x_2x_3 - x_1x_6, x_2x_4 - x_6, x_2x_5 - x_1, x_2x_6 - x_4, $$
$$x_3x_4 - x_1, x_3x_5 - x_6, x_3x_6 - x_5, $$
$$x_4x_5 - x_1x_6, x_4x_6 - x_2, x_5x_6 - x_3 \}. $$
The reduced Gröbner basis of \( I(C) \) with respect to the degree reverse Lexicographical ordering \(<\) is

\[
G = \{ x_1^2 - 1, x_2^2 - 1, x_3^2 - 1, x_4^2 - 1, x_5^2 - 1, x_6^2 - 1, \\
x_1x_2 - x_5, x_1x_3 - x_4, x_1x_4 - x_3, x_1x_5 - x_2, \\
x_2x_3 - x_1x_6, x_2x_4 - x_6, x_2x_5 - x_1, x_2x_6 - x_4, \\
x_3x_4 - x_1, x_3x_5 - x_6, x_3x_6 - x_5, \\
x_4x_5 - x_1x_6, x_4x_6 - x_2, x_5x_6 - x_3 \}. 
\]

Therefore the code is 1-correcting (i.e. \( t = 1 \))
The reduced Gröbner basis of $I(C)$ with respect to the degree reverse Lexicographical ordering $<$ is

$$G = \{ x_1^2 - 1, x_2^2 - 1, x_3^2 - 1, x_4^2 - 1, x_5^2 - 1, x_6^2 - 1, $$
$$x_1x_2 - x_5, x_1x_3 - x_4, x_1x_4 - x_3, x_1x_5 - x_2,$$
$$x_2x_3 - x_1x_6, x_2x_4 - x_6, x_2x_5 - x_1, x_2x_6 - x_4,$$
$$x_3x_4 - x_1, x_3x_5 - x_6, x_3x_6 - x_5,$$
$$x_4x_5 - x_1x_6, x_4x_6 - x_2, x_5x_6 - x_3 \}.$$
1. If we receive \( y = (1, 1, 0, 1, 1, 0) \); then

\[
c_1 := (1, 1, 0, 0, 1, 0) \quad \text{and} \quad y_1 = y + c_1 = (0, 0, 0, 1, 0, 0).
\]

Since \( d(y_1, 0) = 1 \), i.e. codeword corresponding to \( y \) is \( c_1 \).
1. If we receive \( y = (1, 1, 0, 1, 1, 0) \); then

\[
c_1 := (1, 1, 0, 0, 1, 0) \text{ and } y_1 = y + c_1 = (0, 0, 0, 1, 0, 0).
\]

Since \( d(y_1, 0) = 1 \), i.e. codeword corresponding to \( y \) is \( c_1 \).

2. Let \( y = (1, 1, 0, 1, 0, 0) \); then

\[
c_1 := (0, 1, 0, 1, 0, 1) \text{ and } y_1 = y + c_1 = (1, 0, 0, 0, 1, 1).
\]

\( y_1 \) can not be reduced following the algorithm; thus, \( d(y_1, 0) > 1 \); and in this case \( y \) contains more errors than the error-correcting capability of the code. However, note that \( c_1 \) is the closest codeword to \( y \).
The syzygy module and FGLM

Given a set $F = \{f_1, f_2, \ldots, f_r\}$ of polynomials in $K[X] = K[x_1, \ldots, x_n]$ generating an ideal $I$ let us compute a basis for the syzygy module $M$ in $K[X]^{r+1}$ of the generator set $F' = \{-1, f_1, f_2, \ldots, f_r\}$. Each of the syzygies corresponds to a solution

$$f = \sum_{i=1}^{r} b_if_i \quad b_i \in K[X], \quad i = 1, \ldots, r$$

and thus points to an element $f$ in the ideal $I$ generated by $F$. 
The main idea is that the set

$$
\begin{align*}
\mathbf{f}_1 &= (f_1, 1, 0, 0, \ldots, 0) \\
\mathbf{f}_2 &= (f_2, 0, 1, 0, \ldots, 0) \\
\vdots \\
\mathbf{f}_r &= (f_r, 0, 0, 0, \ldots, 1)
\end{align*}
$$

(5)

is a basis of the syzygy module $M$ and moreover it is a Gröbner basis with respect to a position over term (POT) ordering $<_w$ induced from an ordering $<$ in $K[X]$ and the weight vector $\mathbf{w} = (1, T< (f_1), \ldots, T< (f_r))$. 
Now we use the FGLM idea and run through the terms of $K[X]^{r+1}$ in the order determined by $<$ and $u^{(i)} < u^{(j)}$ if $i < j$, using a term over position (TOP) ordering. At each step the canonical form of the term with respect to the original basis is 0 apart from the first component so the determination of the linear relations takes place in that component. This provides a convenient representation for the canonical form with respect to the initial Gröbner basis as a $K$-vector space, and any linear relation obtained as a consequence of reduction of the first component in $K[X]$ will give a corresponding relation for the elements of the module.

**Remark**
Note that the idea of computing a Gröbner basis via the syzygy module can be attributed to Caboara and Traverso.
Now we use the FGLM idea and run through the terms of $K[X]^{r+1}$ in the order determined by $<$ and $u^{(i)} < u^{(j)}$ if $i < j$, using a term over position (TOP) ordering. At each step the canonical form of the term with respect to the original basis is 0 apart from the first component so the determination of the linear relations takes place in that component. This provides a convenient representation for the canonical form with respect to the initial Gröbner basis as a $K$-vector space, and any linear relation obtained as a consequence of reduction of the first component in $K[X]$ will give a corresponding relation for the elements of the module.

**Remark**
Note that the idea of computing a Gröbner basis via the syzygy module can be attributed to Caboara and Traverso.
Toy Example

Consider the binary code $C$ with generator matrix

$$G = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}.$$  

We find that

$$I(C) = \langle f_1 = x_1x_3 - 1, f_2 = x_2x_3 - 1, $$

$$f_3 = x_1^2 - 1, f_4 = x_2^2 - 1, f_5 = x_3^2 - 1 \rangle.$$  

In the associated syzygy computation the rows corresponding to the binomials $x_i^2 - 1$ are considered as implicit in the computations: see, for example, in the Table below when the syzygy corresponding to $x_3 - x_1$ is obtained.
### Worked Example

#### Computing the Gröbner basis

<table>
<thead>
<tr>
<th>multiples of $x_1^2$</th>
<th>$-1$</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_1x_2$</th>
<th>$x_1x_3$</th>
<th>$x_2x_3$</th>
<th>$x_1x_2x_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(1, 0, 0, 0, 0)$</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(1, 1, 0, 0, 0)$</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(1, 0, 1, 0, 0)$</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Thus $x_2 - x_1 = x_1 f_2 - x_2 f_1$, $x_2 - x_1$ belongs to the Gröbner basis and we can now omit all the multiples of $(1, 1, 0, 0, 0)$ from our computation.
Linear codes
Minimal subsets in codes
The Gröbner test set-(trial set)
Worked example
Computing the Groebner basis

### Worked example

#### Computing the Groebner basis

- \( x_1 \)
- \( x_2 \)
- \( x_3 \)
- \( x_1 x_2 \)
- \( x_1 x_3 \)
- \( x_2 x_3 \)
- \( x_1 x_2 x_3 \)

<table>
<thead>
<tr>
<th>( (1, 0, 0, 0, 0, 0) )</th>
<th>( -1 )</th>
<th>( x_1 )</th>
<th>( x_2 )</th>
<th>( x_3 )</th>
<th>( x_1 x_2 )</th>
<th>( x_1 x_3 )</th>
<th>( x_2 x_3 )</th>
<th>( x_1 x_2 x_3 )</th>
<th>multiples of ( x_i^2 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( (1, 0, 0, 0, 0, 0) )</td>
<td>( 1 )</td>
<td>1</td>
<td>( x_2 )</td>
<td>( x_3 )</td>
<td>( x_1 x_2 )</td>
<td>( x_1 x_3 )</td>
<td>( x_2 x_3 )</td>
<td>( x_1 x_2 x_3 )</td>
<td>( x_1^2 )</td>
</tr>
<tr>
<td>( (1, 1, 0, 0, 0, 0) )</td>
<td>( 1 )</td>
<td>( x_1 )</td>
<td>( x_2 )</td>
<td>( x_3 )</td>
<td>( x_1 x_2 )</td>
<td>( x_1 x_3 )</td>
<td>( x_2 x_3 )</td>
<td>( x_1 x_2 x_3 )</td>
<td>( x_1^2 )</td>
</tr>
<tr>
<td>( (1, 0, 1, 0, 0, 0) )</td>
<td>( 1 )</td>
<td>( x_1 )</td>
<td>( x_2 )</td>
<td>( x_3 )</td>
<td>( x_1 x_2 )</td>
<td>( x_1 x_3 )</td>
<td>( x_2 x_3 )</td>
<td>( x_1 x_2 x_3 )</td>
<td>( x_1^2 )</td>
</tr>
</tbody>
</table>

Thus \( x_2 - x_1 = x_1 f_2 - x_2 f_1 \), \( x_2 - x_1 \) belongs to the Gröbner basis and we can now omit all the multiples of \( x_i^2 \) from our computation.
**Linear codes**

**Minimal subsets in codes**

**The Gröbner test set-(trial set)**

**Worked example**

**Computing the Groebner basis**

<table>
<thead>
<tr>
<th></th>
<th>$-1$</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_1x_2$</th>
<th>$x_1x_3$</th>
<th>$x_2x_3$</th>
<th>$x_1x_2x_3$</th>
<th>multiples of $x_i^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(1, 0, 0, 0, 0, 0)$</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(1, 1, 0, 0, 0, 0)$</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(1, 0, 1, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

introduce $x_1$

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$(x_1, 0, 0, 0, 0, 0)$</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(x_1, x_1, 0, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(x_1, 0, x_1, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

introduce $x_2$

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>$(x_2, 0, 0, 0, 0)$</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(x_2, x_2, 0, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(x_2, 0, x_2, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Thus $x_2 - x_1$ belongs to the Gröbner basis and we can now omit all the multiples of $x_i^2$ from our computation.
### Worked Example: Computing the Groebner basis

<table>
<thead>
<tr>
<th></th>
<th>$-1$</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_1x_2$</th>
<th>$x_1x_3$</th>
<th>$x_2x_3$</th>
<th>$x_1x_2x_3$</th>
<th>multiples of $x_i^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(1, 0, 0, 0, 0, 0)$</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(1, 1, 0, 0, 0, 0)$</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(1, 0, 1, 0, 0, 0)$</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>introduce $x_1$</strong></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$x_1^2x_3$</td>
</tr>
<tr>
<td>$(x_1, 0, 0, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(x_1, x_1, 0, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(x_1, 0, x_1, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>introduce $x_2$</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(x_2, 0, 0, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$x_2^2x_3$</td>
</tr>
<tr>
<td>$(x_2, x_0, 0, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(x_2, 0, x_2, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>reduction</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(x_2, 0, 0, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(x_2 - x_1, x_0, x_1, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(x_2, 0, x_2, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$x_2^2x_3$</td>
</tr>
</tbody>
</table>

Thus, $x_2 - x_1$ belongs to the Gröbner basis and we can now omit all the multiples of $x_i^2$ from our computation.
Thus $x_2 - x_1 = x_1f_2 - x_2f_1$, $x_2 - x_1$ belongs to the Gröbner basis and we can now omit all the multiples of $x_2(1, 1, 0, 0, 0, 0, 0)$ from our computation.
Continuing we find

<table>
<thead>
<tr>
<th>introduce $x_3$</th>
<th>$1$</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_1x_2$</th>
<th>$x_1x_3$</th>
<th>$x_2x_3$</th>
<th>$x_1x_2x_3$</th>
<th>multiples of $x_i^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$(x_3, 0, 0, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$x_1x_3^2$</td>
</tr>
<tr>
<td>$(x_3, x_3, 0, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$x_2x_3^2$</td>
</tr>
<tr>
<td>$(x_3, 0, x_3, 0, 0, 0)$</td>
<td></td>
<td>$1$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>reduction $x_3 - x_1$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(x_3, 0, 0, 0, 0, 0)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(x_3 - x_1, x_3, 0, 0, 0, x_1)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$(x_3 - x_2, 0, x_3, 0, 0, x_2)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

We have the syzygy $x_3 - x_1 = x_1f_5 - x_3f_1$ and $x_3 - x_1$ belongs to the Gröbner basis (note that we can make reductions of terms $T \cdot x_i^2$, $T$ a term, as soon as we have introduced $T$ since $x_i^2 - 1$ is a generator). The result of the computation is the Gröbner basis \$\{x_2 - x_1, x_3 - x_1, x_1^2 - 1\}$. 
Thanks for your attention!!