Gröbner bases and Integer Programming

RANT

Alex Dunbar
March 23 2021

Emory University
Table of contents

1. Integer Programs

2. Gröbner Bases

3. Using Gröbner Bases for Integer Programming
Integer Programs
An integer program is an optimization problem

\[
\text{max } c^T x \\
\text{s.t. } Ax = b \\
x \in \mathbb{Z}_+^n
\]
Theorem (Fundamental Theorem of Applied Math)

If $\mathcal{P}$ is a hard problem, then turning $\mathcal{P}$ into a linear algebra problem and throwing it into MATLAB provides a solution.

Proof.

Attend a scientific computing talk.
Traditional Solution of IPs (Serious)

Theorem (Fundamental Theorem of Integer Programming\(^1\))

Let \( \mathcal{X} = \{ x \in \mathbb{Z}^n_+ | Ax = b \} \) the feasible region to (IP). Then,

\[
\max \{ c^T x | x \in \mathcal{X} \} = \max \{ c^T x | x \in \text{conv}(\mathcal{X}) \}
\]

\(^1\)See e.g. Nemhauser-Wolsey *Integer and Combinatorial Optimization*
Theorem (Fundamental Theorem of Integer Programming\textsuperscript{1})

Let $\mathcal{X} = \{x \in \mathbb{Z}^n_+ | Ax = b\}$ the feasible region to (IP). Then,

$$\max\{c^T x | x \in \mathcal{X}\} = \max\{c^T x | x \in \text{conv}(\mathcal{X})\}$$

Key idea: Can solve a sequence of linear programs over polyhedra which contain $\text{conv}(\mathcal{X})$.

\textsuperscript{1}See e.g. Nemhauser-Wolsey *Integer and Combinatorial Optimization*
Theorem (Fundamental Theorem of Integer Programming\(^1\))

Let \( \mathcal{X} = \{x \in \mathbb{Z}_+^n | Ax = b \} \) the feasible region to (IP). Then,

\[
\max \{ c^T x | x \in \mathcal{X} \} = \max \{ c^T x | x \in \text{conv}(\mathcal{X}) \}
\]

Key idea: Can solve a sequence of linear programs over polyhedra which contain \( \text{conv}(\mathcal{X}) \).

Leads to Branch & Bound and Cutting Plane algorithms.

\(^1\)See e.g. Nemhauser-Wolsey *Integer and Combinatorial Optimization*
Example

\[
\begin{align*}
\text{max} & \quad \sum_{i,j,k=1}^{9} x_{i,j,k} \\
\text{s.t.} & \quad \sum_{i=1}^{9} x_{i,j,k} = 1 & \forall j, k, \\
& \quad \sum_{j=1}^{9} x_{i,j,k} = 1 & \forall i, k, \\
& \quad \sum_{(i,j) \in S_\ell} x_{i,j,k} = 1 & \forall k, S_\ell, \\
& \quad \sum_{k=1}^{9} x_{i,j,k} = 1 & \forall i, j, \\
& \quad x_{i,j,k} = 1 & (i, j, k) \in C \\
& \quad x \in \{0, 1\}^{729}
\end{align*}
\]
Example

\[
\begin{align*}
\text{max} \quad & \sum_{i,j,k=1}^{9} x_{i,j,k} \\
\text{s.t.} \quad & \sum_{i=1}^{9} x_{i,j,k} = 1 \quad \forall j, k, \\
& \sum_{j=1}^{9} x_{i,j,k} = 1 \quad \forall i, k, \\
& \sum_{(i,j) \in S_\ell} x_{i,j,k} = 1 \quad \forall k, S_\ell, \\
& \sum_{k=1}^{9} x_{i,j,k} = 1 \quad \forall i, j, \\
& x_{i,j,k} = 1 \quad (i,j,k) \in C \\
x \in \{0,1\}^{729}
\end{align*}
\]
Gröbner Bases
Consider a linear system of equations $Ax = b$, $A \in GL_n(\mathbb{C})$. 
Motivation

Consider a linear system of equations $Ax = b$, $A \in GL_n(\mathbb{C})$.

If $A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,2} & \cdots & a_{2,n} \\ \vdots & \ddots & \vdots \\ a_{n,n} \end{bmatrix}$ is upper triangular, we can solve $Ax = b$ with $n$ equations of a single variable via backsubstitution.
Motivation

Consider a linear system of equations $Ax = b$, $A \in GL_n(\mathbb{C})$.

If $A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,2} & \cdots & a_{2,n} \\ \vdots & \ddots & \vdots \\ a_{n,n} \end{bmatrix}$ is upper triangular, we can solve $Ax = b$ with $n$ equations of a single variable via backsubstitution.

Moreover, we can decompose general $A \in GL_n(\mathbb{C})$ into $A = LU$ and solve $Ax = b$ “easily”.
Fix the lexicographic monomial ordering on $\mathbb{C}[x_1, x_2, \ldots, x_n]$. If $I \subseteq \mathbb{C}[x_1, x_2, \ldots, x_n]$ is an ideal, set $LT(I) = \langle LT(f) \mid f \in I \rangle$.

\[\text{See e.g Dummit& Foote Section 9.6 or Cox, Little, O'Shea }Ideals, Varieties, and Algorithms Ch. 2& 3]
Fix the lexicographic monomial ordering on $\mathbb{C}[x_1, x_2, \ldots, x_n]$. If $I \subseteq \mathbb{C}[x_1, x_2, \ldots, x_n]$ is an ideal, set $LT(I) = \langle LT(f) | f \in I \rangle$.

A subset $G \subseteq I$ is a Gröbner Basis of $I$ if $I = \langle G \rangle$ and $LT(I) = \langle LT(g) | g \in G \rangle$.

---

\[ ^2 \text{See e.g Dummit& Foote Section 9.6 or Cox, Little, O’Shea } \text{Ideals, Varieties, and Algorithms Ch. 2& 3} \]
Fix the lexicographic monomial ordering on \( \mathbb{C}[x_1, x_2, \ldots, x_n] \). If \( I \subseteq \mathbb{C}[x_1, x_2, \ldots, x_n] \) is an ideal, set \( LT(I) = \langle LT(f) | f \in I \rangle \).

A subset \( G \subseteq I \) is a Gröbner Basis of \( I \) if \( I = \langle G \rangle \) and \( LT(I) = \langle LT(g) | g \in G \rangle \).

\( G \) is reduced if each \( g_i \in G \) is monic and if \( LT(g_i) \) does not divide any term of \( g_j \) for \( i \neq j \).

---

\(^2\)See e.g Dummit& Foote Section 9.6 or Cox, Little, O’Shea *Ideals, Varieties, and Algorithms* Ch. 2& 3
Gröbner Bases

Theorem

1. Every ideal \( I \subset \mathbb{C}[x_1, x_2, \ldots, x_n] \) has a unique reduced Gröbner basis \( G = \{g_1, g_2, \ldots, g_m\} \)

2. Every \( f \in \mathbb{C}[x_1, x_2, \ldots, x_n] \) can be written uniquely in the form \( f = f_i + r \), where \( f_i \in I \) and no monomial in \( r \) is divisible by \( \text{LT}(g_i) \).

Denote \( r = f \mod G \).
Given \( I = \langle f_1, f_2, \ldots, f_m \rangle \subseteq \mathbb{C}[x_1, x_2, \ldots, x_n] \), we can find a Gröbner Basis:

1. \( G = \{ f_1, f_2, \ldots, f_m \} \)
Buchberger’s Algorithm

Given \( I = \langle f_1, f_2, \ldots, f_m \rangle \subseteq \mathbb{C}[x_1, x_2, \ldots, x_n] \), we can find a Gröbner Basis:

1. \( G = \{f_1, f_2, \ldots, f_m\} \)

2. For each \( f_i, f_j \in G \),
   2.1 Set \( S(f_i, f_j) = \frac{\text{LCM}(\text{LT}(f_i), \text{LT}(f_j))}{\text{LT}(f_i)} \frac{f_i}{f_i} - \frac{\text{LCM}(\text{LT}(f_i), \text{LT}(f_j))}{\text{LT}(f_j)} \frac{f_j}{f_j} \)
Buchberger’s Algorithm

Given \( I = \langle f_1, f_2, \ldots, f_m \rangle \subseteq \mathbb{C}[x_1, x_2, \ldots, x_n] \), we can find a Gröbner Basis:

1. \( G = \{ f_1, f_2, \ldots, f_m \} \)
2. For each \( f_i, f_j \in G \),
   2.1 Set \( S(f_i, f_j) = \frac{\text{LCM}(\text{LT}(f_i), \text{LT}(f_j))}{\text{LT}(f_i)} f_i - \frac{\text{LCM}(\text{LT}(f_i), \text{LT}(f_j))}{\text{LT}(f_j)} f_j \)
   2.2 If \( S(f_i, f_j) \equiv 0 \mod G \), continue iterating
Buchberger’s Algorithm

Given \( I = \langle f_1, f_2, \ldots, f_m \rangle \subseteq \mathbb{C}[x_1, x_2, \ldots, x_n] \), we can find a Gröbner Basis:

1. \( G = \{ f_1, f_2, \ldots, f_m \} \)

2. For each \( f_i, f_j \in G \),
   
   2.1 Set \( S(f_i, f_j) = \frac{\text{LCM}(\text{LT}(f_i), \text{LT}(f_j))}{\text{LT}(f_i)}f_i - \frac{\text{LCM}(\text{LT}(f_i), \text{LT}(f_j))}{\text{LT}(f_j)}f_j \)
   
   2.2 If \( S(f_i, f_j) = 0 \mod G \), continue iterating
   
   2.3 If \( S(f_i, f_j) \neq 0 \mod G \), append \( G = G \cup \{ S(f_i, f_j) \mod G \} \) and go back to 2.
Example

Consider \( I = \langle x_1 + x_2 + 3x_3 - 4, x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3 \rangle \).
Consider $I = \langle x_1 + x_2 + 3x_3 - 4, x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3 \rangle$.

A Gröbner Basis for $I$ is

$$G = \left\{ x_1 + x_2 + 3x_3 - 4, x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3, 
\quad x_2x_3 - x_2 - 2x_3 + 2, x_3 - 1 \right\}$$
Consider $I = \langle x_1 + x_2 + 3x_3 - 4, x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3 \rangle$.

A Gröbner Basis for $I$ is

$$G = \left\{ x_1 + x_2 + 3x_3 - 4, x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3, 
    x_2x_3 - x_2 - 2x_3 + 2, x_3 - 1 \right\}$$

The reduced Gröbner basis for $I$ is

$$G_{\text{red}} = \left\{ x_1 + x_2 - 1, x_2^2 - x_2, x_3 - 1 \right\}$$
For each $1 \leq \ell \leq n$, let $I_\ell = I \cap \mathbb{C}[x_{\ell+1}, x_{\ell+2}, \ldots, x_n]$ be the $\ell$th elimination ideal.
For each $1 \leq \ell \leq n$, let $I_\ell = I \cap \mathbb{C}[x_{\ell+1}, x_{\ell+2}, \ldots, x_n]$ be the $\ell^{th}$ elimination ideal.

**Theorem**

If $G$ is a Gröbner basis of $I$, then $G \cap \mathbb{C}[x_{\ell+1}, x_{\ell+2}, \ldots, x_n]$ is a Gröbner basis of $I_\ell$. 
Why is elimination useful?
Why is elimination useful?

Theorem

Let \( I = \langle f_1, f_2, \ldots, f_s \rangle \subseteq \mathbb{C}[x_1, x_2, \ldots, x_n] \). Then,
\[ \text{LT}(f_i) = g_i(x_2, x_3, \ldots, x_n)x_1^{N_i} \]
for some \( g_i \in \mathbb{C}[x_2, x_3, \ldots, x_n] \). Moreover, if \( (a_2, a_3, \ldots, a_n) \in V(I_1) \), and \( (a_2, a_3, \ldots, a_n) \not\in V(\langle g_i \rangle) \), then \( \exists a_1 \in \mathbb{C} \) such that \( (a_1, a_2, \ldots, a_n) \in V(I) \).
Why is elimination useful?

**Theorem**

Let \( I = \langle f_1, f_2, \ldots, f_s \rangle \subseteq \mathbb{C}[x_1, x_2, \ldots, x_n] \). Then, \( \text{LT}(f_i) = g_i(x_2, x_3, \ldots, x_n)x_1^{N_i} \) for some \( g_i \in \mathbb{C}[x_2, x_3, \ldots, x_n] \). Moreover, if \( (a_2, a_3, \ldots, a_n) \in V(I_1) \), and \( (a_2, a_3, \ldots, a_n) \notin V(\langle g_i \rangle) \), then \( \exists a_1 \in \mathbb{C} \) such that \( (a_1, a_2, \ldots, a_n) \in V(I) \).

Key takeaway: Can use partial solutions to get full solutions.
We had $G = \{ x_1 + x_2 - 1, x_2^2 - x_2, x_3 - 1 \}$ was a reduced Gröbner basis of $I = \langle x_1 + x_2 + 3x_3 - 4, x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3 \rangle$. 
We had \( G = \{ x_1 + x_2 - 1, x_2^2 - x_2, x_3 - 1 \} \) was a reduced Gröbner basis of \( I = \langle x_1 + x_2 + 3x_3 - 4, x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3 \rangle \).

\[
G_2 = G \cap \mathbb{C}[x_3] = \langle x_3 - 1 \rangle \implies \left[ x \in V(I_2) \implies x_3 = 1 \right]
\]
We had \( G = \{ x_1 + x_2 - 1, x_2^2 - x_2, x_3 - 1 \} \) was a reduced Gröbner basis of \( I = \langle x_1 + x_2 + 3x_3 - 4, x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3 \rangle \).

\[
G_2 = G \cap \mathbb{C}[x_3] = \langle x_3 - 1 \rangle \implies [x \in V(I_2) \implies x_3 = 1]
\]

\[
l_1 = \langle x_2^2 - x_2, x_3 - 1 \rangle \implies [x \in V(l_1) \implies x_2 \in \{0, 1\} \text{ and } x_3 = 1]
\]
We had \( G = \{ x_1 + x_2 - 1, x_2^2 - x_2, x_3 - 1 \} \) was a reduced Gröbner basis of \( I = \langle x_1 + x_2 + 3x_3 - 4, x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3 \rangle \).

\[
G_2 = G \cap \mathbb{C}[x_3] = \langle x_3 - 1 \rangle \implies [x \in V(I_2) \implies x_3 = 1]
\]

\[
I_1 = \langle x_2^2 - x_2, x_3 - 1 \rangle \implies [x \in V(I_1) \implies x_2 \in \{0, 1\} \text{ and } x_3 = 1]
\]

\[
I = \langle x_1 + x_2 - 1, x_2^2 - x_2, x_3 - 1 \rangle \implies V(I) = \{(1, 0, 1), (0, 1, 1)\}
\]
Using Gröbner Bases for Integer Programming
Restrict our attention to binary integer programs:

\[ \mathcal{X} = \{ x \in \{0, 1\}^n | Ax = b \} \]
Restrict our attention to binary integer programs:

\[ \mathcal{X} = \{ x \in \{0, 1\}^n | Ax = b \} \]

Then,

\[ x \in \mathcal{X} \iff x \in V(\langle x_i^2 - x_i, \sum_{j=1}^{n} a_{i,j}x_j - b_i \rangle) \]

\[ ^3 \text{Bertsimas et. al A New Algebraic Geometry Algorithm for Integer Programming} \]
Solving IPs

Using

\[ x \in \mathcal{X} \iff x \in V(\langle x_i^2 - x_i, \sum_{j=1}^{n} a_{i,j}x_j - b_i \rangle) \]

and the Elimination and Extension theorems, we can either determine that a problem is infeasible or enumerate its solutions.
Example Part 3

Consider the BIP feasible region

\[ \mathcal{X} = \{ x \in \{0, 1\}^3 \mid x_1 + x_2 + 3x_3 = 4 \} \]
Consider the BIP feasible region

\[ \mathcal{X} = \{ x \in \{0, 1\}^3 | x_1 + x_2 + 3x_3 = 4 \} \]

In our “new” language,

\[ \mathcal{X} = V(\langle x_1 + x_2 + 3x_3 - 4, x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3 \rangle) \]
Consider the BIP feasible region

\[ \mathcal{X} = \{ x \in \{0, 1\}^3 \mid x_1 + x_2 + 3x_3 = 4 \} \]

In our “new” language,

\[ \mathcal{X} = V(\langle x_1 + x_2 + 3x_3 - 4, x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3 \rangle) \]
\[ = V(\langle x_1 + x_2 - 1, x_2^2 - x_2, x_3 - 1 \rangle) \]
Consider the BIP feasible region

\[ \mathcal{X} = \{ x \in \{0, 1\}^3 | x_1 + x_2 + 3x_3 = 4 \} \]

In our “new” language,

\[ \mathcal{X} = V(\langle x_1 + x_2 + 3x_3 - 4, x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3 \rangle) \]
\[ = V(\langle x_1 + x_2 - 1, x_2^2 - x_2, x_3 - 1 \rangle) \]
\[ = \{(1, 0, 1), (0, 1, 1)\} \]
The terms of the reduced Göbner basis capture the logical dependency of the optimization variables:

$$\langle x_1 + x_2 - 1, x_2^2 - x_2, x_3 - 1 \rangle$$
The terms of the reduced Góbner basis capture the logical dependency of the optimization variables:

$$\langle x_1 + x_2 - 1, x_2^2 - x_2, x_3 - 1 \rangle$$

Exactly one of $x_1$ or $x_2$ is 1, $x_3$ must be 1.
Add in an objective function $z = c^T x$:

$$\begin{align*}
\text{max} & \quad z = c^T x \\
\text{s.t.} & \quad Ax = b \\
& \quad x \in \{0, 1\}^n
\end{align*}$$
Maximization

Add in an objective function $z = c^\top x$:

$$\max \ z = c^\top x$$

s.t. $Ax = b$

$x \in \{0, 1\}^n$

Look at

$$l = \langle x_i^2 - x_i, \sum_{j=1}^n a_{i,j}x_j - b_i, \sum_{j=1}^n c_jx_j - z \rangle \subseteq \mathbb{C}[x_1, x_2, \ldots, x_n, z]$$
Example Part 4

Look at the Binary IP

\[
\begin{align*}
\text{max} & \quad z = x_1 + 3x_2 + 5x_3 \\
\text{s.t.} & \quad x_1 + x_2 + 3x_3 = 4 \\
& \quad x \in \{0, 1\}^3
\end{align*}
\]
Example Part 4

Look at the Binary IP

\[
\begin{align*}
\text{max} & \quad z = x_1 + 3x_2 + 5x_3 \\
\text{s.t.} & \quad x_1 + x_2 + 3x_3 = 4 \\
& \quad x \in \{0, 1\}^3
\end{align*}
\]

Ideal is

\[
l = \langle x_1 + x_2 + 3x_3 - 4, x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3, x_1 + 3x_2 + 5x_3 - z \rangle
\]
Look at the Binary IP

\[
\begin{align*}
\text{max} \quad & z = x_1 + 3x_2 + 5x_3 \\
\text{s.t.} \quad & x_1 + x_2 + 3x_3 = 4 \\
& x \in \{0, 1\}^3
\end{align*}
\]

Ideal is

\[
I = \langle x_1 + x_2 + 3x_3 - 4, x_1^2 - x_1, x_2^2 - x_2, x_3^2 - x_3, x_1 + 3x_2 + 5x_3 - z \rangle \\
= \langle x_1 + \frac{1}{2}z - 4, x_2 - \frac{1}{2}z + 3, x_3 - 1, z^2 - 14z + 48 \rangle
\]
Example Part 4 (cont.)

\[ l = \langle x_1 + \frac{1}{2}z - 4, x_2 - \frac{1}{2}z + 3, x_3 - 1, z^2 - 14z + 48 \rangle \]

So, \( l_3 = l \cap \mathbb{C}[z] = \langle z^2 - 14z + 48 \rangle \).
Example Part 4 (cont.)

\[ I = \langle x_1 + \frac{1}{2}z - 4, x_2 - \frac{1}{2}z + 3, x_3 - 1, z^2 - 14z + 48 \rangle \]

So, \[ l_3 = I \cap \mathbb{C}[z] = \langle z^2 - 14z + 48 \rangle. \]

Feasible Objective Values are:

\[ V(l_3) = \{6, 8\} \]
Example Part 4 (cont.)

\[ I = \langle x_1 + \frac{1}{2}z - 4, x_2 - \frac{1}{2}z + 3, x_3 - 1, z^2 - 14z + 48 \rangle \]

So, \( l_3 = I \cap \mathbb{C}[z] = \langle z^2 - 14z + 48 \rangle \).

Feasible Objective Values are:

\[ V(l_3) = \{ 6, 8 \} \]

Take \( z = 8 \) and apply extension theorem:

\[ (x_1, x_2, x_3; z) = (0, 1, 1; 8) \]

is optimal solution & objective value.
• Integer Programs are an interesting class of optimization problems
• Gröbner Bases assist in computation in polynomial rings
• Integer Programs can be translated into questions about ideals in $\mathbb{C}[x_1, x_2, \ldots, x_n, z]$.
• Don’t try to implement this in MATLAB