Original Buchberger-Möller (BM) Algorithm

Given:

- a (finite!) set of points $\mathbb{X} \subseteq K^n$
- a term-ordering σ

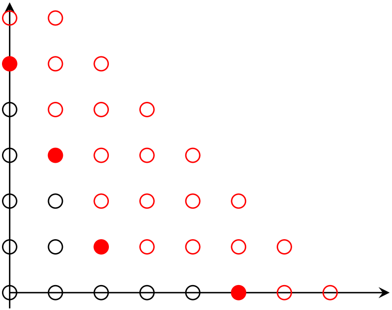
Compute:

- **Reduced Gröbner Basis** (RGB) of the associated ideal
- byproduct: *separator polynomials* satisfying $f_i(P_j) = \delta_{ij}$

Structure of Reduced Gröbner Basis (RGB)

Salient structural features of RGB of 0-dim ideal I

PP diagram



- ○ Quotient Basis (QB) element
- ● LT of RGB element
- ○ other element of LT(I)

- QB $\longleftrightarrow$ vec.sp. basis of P/I
  $\Longrightarrow \quad \sum_{t \in QB} c_t t \in I \quad \Longleftrightarrow \quad$ **all $c_t = 0$**
- RGB element has form $g_j = t_j - \sum_{t \in QB} c_t t$ using only $t < t_j$

Original BM Algorithm: idea

Basic idea: build QB one element at a time in **increasing order**:

$$QB_1 = \{t_1 = 1\}$$
$$QB_2 = QB_1 \cup \{t_2\}$$
$$QB_3 = QB_2 \cup \{t_3\}$$
$$QB_4 = QB_3 \cup \{t_4\}$$

and so on…

…until we discover a lin.comb. $g = t_k - \sum_{t \in QB_{k-1}} c_t t \quad \in I$

$\longrightarrow \quad g$ is an **element of RGB**.

Keep going until all of RGB has been found!

Original BM Algorithm: evaluation matrix

Qn: How to tell when there is a lin.comb. $g = t_k - \sum_{t \in QB_{k-1}} c_t t \quad \in I$?

Consider the **evaluation matrix**:

$$M = \begin{pmatrix} t_1(P_1) & t_1(P_2) & \cdots & t_1(P_s) \\ t_2(P_1) & t_2(P_2) & \cdots & t_2(P_s) \\ \vdots & \vdots & & \vdots \\ t_k(P_1) & t_k(P_2) & \cdots & t_k(P_s) \end{pmatrix}$$

Let r_i denote the i-th row of M.

$\longrightarrow$ entries are the values of the polynomial t_i at the points $P_1, \ldots, P_s$.

Let $f = \sum_{i=1}^k a_i t_i$ be any polynomial.

Evaluation is a homomorphism

$\Longrightarrow$ the lin.comb. of rows $\sum_{i=1}^k a_i r_i$ has j-th coord $\sum a_i t_i(P_j) = f(P_j)$

$$f \in I \quad \Longleftrightarrow \quad f(P_j) = 0 \quad \forall j \quad \Longleftrightarrow \quad \sum a_i r_i = 0$$

Ans: there is $g \in I$ iff **last row of M is lin.dep. on the other rows**!

Original BM Algm: running example

PP diagram

Start with PP $t_1 = 1$.

$$M = \begin{pmatrix} t_1(P_1) & t_1(P_2) & \cdots & t_1(P_s) \end{pmatrix}$$

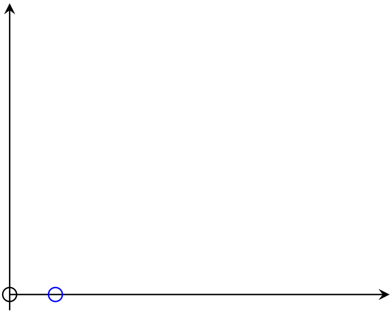
First row is just $(1, 1, \ldots, 1)$, so there is no linear dependency.

$\longrightarrow QB = \{t_1\} \qquad RGB = \{\}$

Proceed to next PP.

Original BM Algm: running example 2

PP diagram



Next PP is $t_2 = x$.

$$M = \begin{pmatrix} t_1(P_1) & t_1(P_2) & \cdots & t_1(P_s) \\ t_2(P_1) & t_2(P_2) & \cdots & t_2(P_s) \end{pmatrix}$$

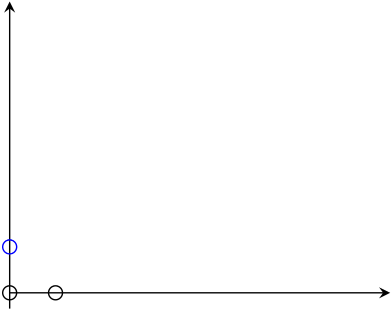
We find no linear dependency.

$\longrightarrow QB = \{t_1, t_2\} \qquad RGB = \{\}$

Consider next PP.

Original BM Algm: running example 3

PP diagram



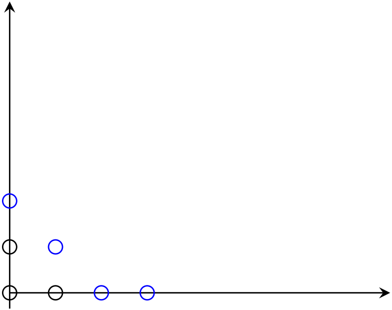
Next PP is $t_3 = y$.

$$M = \begin{pmatrix} t_1(P_1) & t_1(P_2) & \cdots & t_1(P_s) \\ t_2(P_1) & t_2(P_2) & \cdots & t_2(P_s) \\ t_3(P_1) & t_3(P_2) & \cdots & t_3(P_s) \end{pmatrix}$$

We find no linear dependency.
$\longrightarrow QB = \{t_1, t_2, t_3\}$ $\qquad RGB = \{\}$
Consider next PP.

Original BM Algm: running example 4

PP diagram



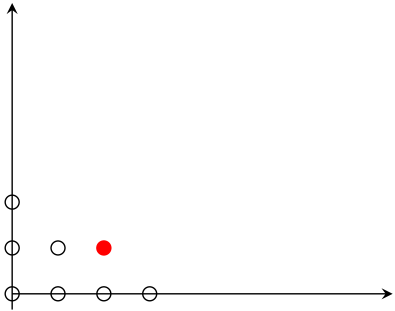
Next PPs are
$t_4 = x^2, t_5 = xy, t_6 = y^2, t_7 = x^3$

$$M = \begin{pmatrix} t_1(P_1) & t_1(P_2) & \cdots & t_1(P_s) \\ t_2(P_1) & t_2(P_2) & \cdots & t_2(P_s) \\ \vdots & \vdots & & \vdots \\ t_7(P_1) & t_7(P_2) & \cdots & t_7(P_s) \end{pmatrix}$$

We still find no linear dependency.
$\longrightarrow QB = \{t_1, \ldots\ldots, t_7\} \qquad RGB = \{\}$
Consider next PP.

Original BM Algm: running example 5

PP diagram

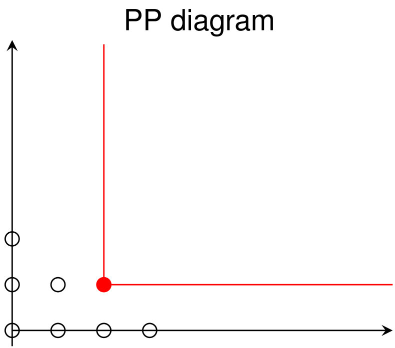


Next PP is $t_8 = x^2 y$

$$M = \begin{pmatrix} t_1(P_1) & t_1(P_2) & \cdots & t_1(P_s) \\ t_2(P_1) & t_2(P_2) & \cdots & t_2(P_s) \\ \vdots & \vdots & & \vdots \\ t_7(P_1) & t_7(P_2) & \cdots & t_7(P_s) \\ t_8(P_1) & t_8(P_2) & \cdots & t_8(P_s) \end{pmatrix}$$

We find a **linear dependency**: $r_8 = \sum_{t \in QB} c_t r_t$
so we get RGB element $g_1 = t_8 - \sum_{t \in QB} c_t t$
$\longrightarrow QB = \{t_1 \ldots, t_7\}$ $\quad RGB = \{g_1\}$

Original BM Algm: running example 6



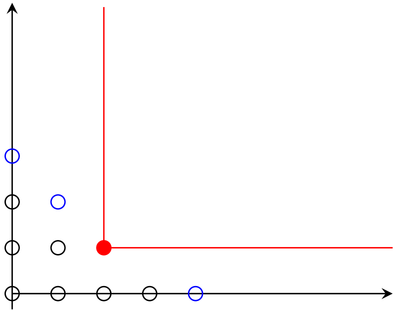
Remove last row from M:

$$M = \begin{pmatrix} t_1(P_1) & t_1(P_2) & \cdots & t_1(P_s) \\ t_2(P_1) & t_2(P_2) & \cdots & t_2(P_s) \\ \vdots & \vdots & & \vdots \\ t_7(P_1) & t_7(P_2) & \cdots & t_7(P_s) \\ \cancel{t_8(P_1)} & \cancel{t_8(P_2)} & \cdots & \cancel{t_8(P_s)} \end{pmatrix}$$

Exclude all multiples of t_8 from further consideration
$\longrightarrow$ red quadrant in diagram.
Proceed to next PP.

Original BM Algm: running example 7

PP diagram



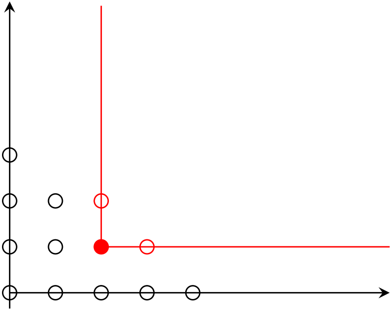
Next PPs are:
$t_9 = xy^2, t_{10} = y^3, t_{11} = x^4$

$$M = \begin{pmatrix} t_1(P_1) & t_1(P_2) & \cdots & t_1(P_s) \\ t_2(P_1) & t_2(P_2) & \cdots & t_2(P_s) \\ \vdots & \vdots & & \vdots \\ t_{11}(P_1) & t_{11}(P_2) & \cdots & t_{11}(P_s) \end{pmatrix}$$

There are no linear relations among the rows.
$\longrightarrow QB = \{t_1, \ldots \ldots, t_{11}\}$ $\quad RGB = \{g_1\}$
Proceed to next PP.

Original BM Algm: running example 8

PP diagram



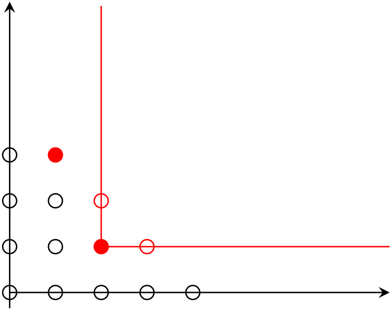
Skipping 2 PPs...

$$M = \begin{pmatrix} t_1(P_1) & t_1(P_2) & \cdots & t_1(P_s) \\ t_2(P_1) & t_2(P_2) & \cdots & t_2(P_s) \\ \vdots & \vdots & & \vdots \\ t_{11}(P_1) & t_{11}(P_2) & \cdots & t_{11}(P_s) \end{pmatrix}$$

We skip x^3y and x^2y^2 as they are excluded.
Consider next PP.

Original BM Algm: running example 9

PP diagram



The next PP is $t_{12} = xy^3$

$$M = \begin{pmatrix} t_1(P_1) & t_1(P_2) & \cdots & t_1(P_s) \\ t_2(P_1) & t_2(P_2) & \cdots & t_2(P_s) \\ \vdots & \vdots & & \vdots \\ t_{11}(P_1) & t_{11}(P_2) & \cdots & t_{11}(P_s) \\ t_{12}(P_1) & t_{12}(P_2) & \cdots & t_{11}(P_s) \end{pmatrix}$$

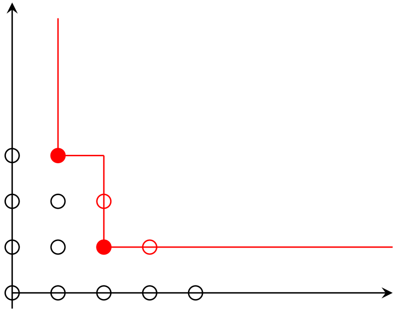
There is a **new linear relation**

$\longrightarrow$ get a new RGB element: $g_2 = t_{12} - \sum_{t \in QB} c_t t$

$\longrightarrow$ $QB = \{t_1, \ldots, t_{11}\}$ $\qquad RGB = \{g_1, g_2\}$

Original BM Algm: running example 10

PP diagram



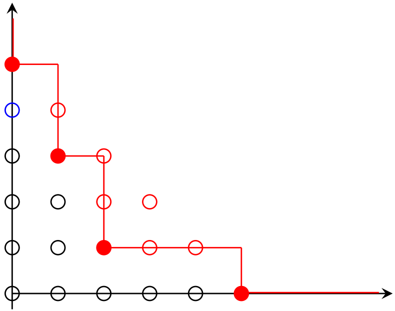
Remove the last row from M:

$$M = \begin{pmatrix} t_1(P_1) & t_1(P_2) & \cdots & t_1(P_s) \\ t_2(P_1) & t_2(P_2) & \cdots & t_2(P_s) \\ \vdots & \vdots & & \vdots \\ t_{11}(P_1) & t_{11}(P_2) & \cdots & t_{11}(P_s) \\ \cancel{t_{12}(P_1)} & \cancel{t_{12}(P_2)} & \cdots & \cancel{t_{12}(P_s)} \end{pmatrix}$$

Exclude all multiples of t_{12} from future consideration
$\longrightarrow$ union of 2 red quadrants in diagram.
Consider next PP.

Original BM Algm: running example 11

PP diagram



Last PPs:

$t_{13} = y^4$, $t_{14} = x^5$, $t_{15} = y^5$

excluded: x^4y, x^3y^2, x^2y^3, xy^4

$$M = \begin{pmatrix} t_1(P_1) & t_1(P_2) & \cdots & t_1(P_s) \\ t_2(P_1) & t_2(P_2) & \cdots & t_2(P_s) \\ \vdots & \vdots & & \vdots \\ t_{13}(P_1) & t_{13}(P_2) & \cdots & t_{13}(P_s) \end{pmatrix}$$

We have found the final two RGB elements g_3 and g_4

$\longrightarrow$ $QB = \{t_1, \ldots, t_{12}, t_{13}\}$ $\quad$ $RGB = \{g_1, g_2, g_3, g_4\}$

Exclude all multiples of x^5 and y^5

$\longrightarrow$ there are no further PPs to consider

$\longrightarrow$ the algorithm terminates.

Original BM Algorithm

(1) initialization: $RGB = \emptyset$, $QB = \emptyset$, $L = \{1\}$, M=$0 \times s$ matrix

(2) While $L \neq \emptyset$ do

- (2a) Set $t = \min_\sigma(L)$ and remove t from L.
- (2b) Compute the **evaluation vector** $v = (t(p_1), \ldots, t(p_s)) \in K^s$
- (2c) **if** v is **linearly dependent** on the rows of M
  then (wlog $v = \sum_i a_i r_i$) add $t - \sum_i a_i QB[i]$ to RGB and remove from L all multiples of t
  else
  add v as a new row to M; add t to QB; and add to L those elements of $\{x_1 t, \ldots, x_n t\}$ which are neither multiples of an element of L nor of $\mathrm{LT}_\sigma(RGB)$.

(3) return (RGB, QB)

Implementation ideas

Ideas for a **good implementation**

- find candidate QB via *fast* computation mod p
  $\longrightarrow$ **structure of answer**
- compute *evaluation matrix* of QB without modulus
- recall that in QB each PP $\neq 1$ is x_j times another PP in QB.
- find coeffs of RGB elements by solving linear system
  $\longrightarrow$ *corners* are LTs of RGB elements.
- if insoluble or soln gives *wrong* LT, try another prime p

XBM: Normal Form Vector

Key idea: **normal form vector** map $NFV : K[x_1, \dots, x_n] \longrightarrow K^m$

NFV needs the following properties:

- $NFV(f) = 0$ iff f is in the ideal
- behave well wrt multiplication

In practice NFV comprises

- a vector $u \in K^m$ such that $NFV(1) = u$
- a collection of commuting multiplication matrices $M_1, \dots, M_n$ such that $NFV(x_j \cdot f) = M_j \cdot NFV(f)$

Observation $NFV_1 \oplus NFV_2$ is also an NFV map $\longleftrightarrow$ intersection

XBM: NFV examples 1

Let I be a zero-dimensional ideal
$\Longrightarrow$ P/I is finite dimensional vector space
$\longrightarrow$ Let $e_1, \ldots, e_m$ be a basis for P/I.

Define $NFV_I : P \longrightarrow K^m$ by $f \mapsto (c_1, \ldots, c_m)$ where $\sum c_i e_i = NF_I(f)$

Example: Simple point *e.g.* $P = (2, 4, 6) \in K^3$
$\longrightarrow$ associated ideal $I = (x - 2, y - 4, z - 6)$
$\longrightarrow$ $\dim(K[x, y, z]/I) = 1$
$\longrightarrow$ NFV_P is just evaluation at P

XBM: NFV examples 2

Recall **original BM** computes intersection $\mathrm{ideal}(P_1) \cap \cdots \cap \mathrm{ideal}(P_s)$

$$NFV : t \mapsto NFV_{P_1}(t) \oplus \cdots \oplus NFV_{P_s}(t) \in K^s$$

XBM extends naturally to ideals of **points with multiplicity**
$\longrightarrow$ **fat points** $\longleftrightarrow$ power $(x_1 - a_1, \ldots, x_n - a_n)^k$
$\longrightarrow$ other **multiple points** such as (x, y^2) or $(x^3, y^5 - x^2)$

Let I be a zero-dimensional ideal with a G-basis, so NFV_I is explicit.

Original BM $+NFV_I =$ **FGLM** $\longleftrightarrow$ **change of ordering** for G-bases.

XBM: NFV examples 3

Let $G = \langle M_1, \ldots, M_n \mid M_i \cdot M_j = M_j \cdot M_i \rangle$ be a commutative semigroup of $s \times s$ matrices.

Define $NFV_G : P \longrightarrow K^s$ by $f \mapsto f(M_1, \ldots, M_n)$ then **flatten** the matrix.

Special case:

if $G = \langle M \rangle$ is cyclic then XBM computes **minimal poly** of M.

XBM: Projective Points

Points in projective space are defined **up to a scalar multiple**
$\Longrightarrow$ seek **homogeneous** polynomials.

Problem: algebraically no longer zero-dimensional!

Compute **degree-by-degree**
$\longrightarrow$ computation in each degree is finite
$\longrightarrow$ but when to stop???

Stopping criterion:

- purely combinatorial
- derived from properties of the Hilbert function

XBM: Factor closedness of QB

Why a **factor closed** QB?

- **obligatory** for RGB (original BM algm)
- **natural** choice $\longrightarrow$ contains the simplest PPs
- QB **remains valid under translation** of the orig points (& scaling)

In general QB does **not remain valid under rotation**: for example...

XBM can build QB not necessarily in increasing order
$\longrightarrow$ but do not have complete freedom.

XBM: Border bases

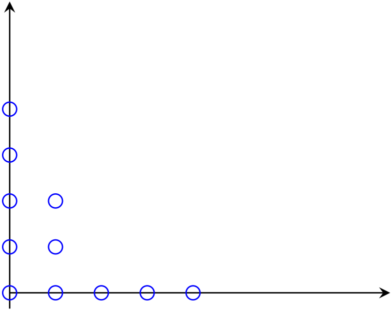
Problem: given affine points, find a border basis for the ideal.

- **structure is determined by QB**
  $\longrightarrow$ QB may be *any* set having $\det(M) \neq 0$
  $\longrightarrow$ QB may be found using BM algorithm, or ...
  $\longrightarrow$ **if** we use BM algm **then** RGB is subset of BB
- LT of each BB element determined by the QB
- coeffs of each BB element can be found via linear algebra
- B-basis is **structurally stable** because determined by an **open condition**

Note: notion of border basis is inherently 0-dimensional

Border basis from QB (1)

PP diagram

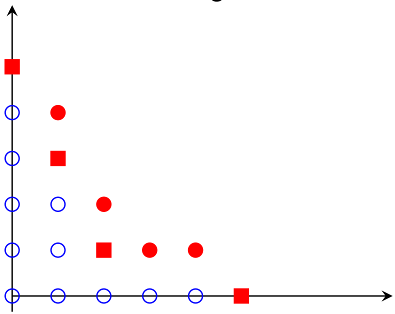


$\circ \leftrightarrow$ quotient basis element

For instance, we can use this QB from the previous running example.

Border basis from QB (2)

PP diagram



■ ↔ **corner BB element**
● ↔ other BB element
○ ↔ QB element

LT of BB are 1 step away from QB.

If QB is **compatible with a term-ordering** then RGB is a subset of BB corresponding to the **corners**.

XBM and Approximate Data (1)

Motivation:

to find **approx polynomial relations** between experimental data

XBM is promising because:

- **guided by linear algebra** (decides if t gives QB element or RGB element)
- **coeffs** of output polys are **found using linear algebra**
- approximate linear algebra is (fairly) well understood

XBM and Approximate Data (2)

What is a *set of approximate points*?

Each approximate point comprises two parts $\tilde{p} = (p, N)$:

- a central exact point p
- a neighbourhood of admissible perturbations $N_\varepsilon(p)$

We additonally assume that the neighbourhoods are:

- identical (up to translation)
- given by $N_\varepsilon(p) = \{x \in \mathbb{R}^n : ||x - p|| < \varepsilon\}$

BM and Approximate Data (3)

Main problem:
$\rightarrow$ given a set of approximate points
$\leftarrow$ find *one or more* approx polynomial relations between them.

Outline of method:

- find a suitable QB using **approx linear dependency** as a guide;
- find the **best polynomial(s)** with given support.

BM and Approximate Data (4)

How to measure **approx linear dependency**?

$\longrightarrow$ small **least squares residual**

$\longrightarrow$ small **singular value**

$\longrightarrow$ other???

What is the **best polynomial**?

$\longrightarrow$ minimise $||\cdot||_2$ of **values** at central points

$\longrightarrow$ minimise $||\cdot||_\infty$ of **values** at central points

$\longrightarrow$ minimise $||\cdot||_2$ of **distances** to central points

$\longrightarrow$ minimise $||\cdot||_\infty$ of **distances** to central points

$\longrightarrow$ other???

XBM and Approximate Data (5)

A **problem** with finding **several** “approximate polynomial relations”

Taken together as a system these polynomials may be a **poor model**:

- different cardinality
- common solutions may be far from the original points

$\longrightarrow$ notion of stable complete intersection

Example: with $\epsilon = 0.005$

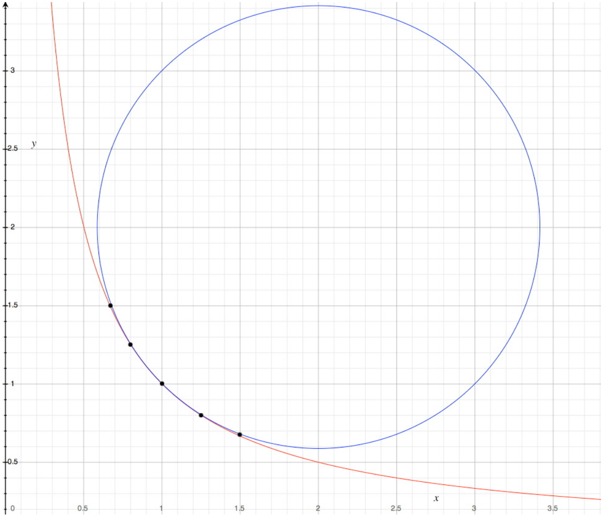
$(0.672, 1.500)$ $(0.800, 1.251)$ $(1.000, 1.000)$ $(1.251, 0.800)$ $(1.500, 0.672)$

$\longrightarrow$ points are **well separated.**

One good polynomial relation is $xy - 1 = 0$
Another good polynomial relation is $(x-2)^2 + (y-2)^2 - 2 = 0$

$\longrightarrow$ they define **two curves that are very similar** over a certain range
$\longrightarrow$ they have just a **single common solution**: $(1, 1)$

BM and Approximate Data (6)



Both curves are a good fit: $xy - 1 = 0$ and $(x-2)^2 + (y-2)^2 - 2 = 0$