Part 2

Part 2 contains:

- Some theory
- A hands on explanation

Please ask questions!

Important Words

- A Monomial is a finite product of unknowns.

$$\mathrm{x}$$
$$\mathrm{xy^5z^3}$$

- A Polynomial is a finite sum of monomials.

$$\mathrm{4x^5 + 3x^3 + 1}$$
$$\mathrm{xyz^5 + 7x^2y + 1}$$

Ideal

The ideal of $\{f_i\}$ is the set

$$I(\{f_i\}) = \left\{ \sum_{\text{finite}} f_j p_j \mid f_j \in \{f_i\}, p_j \in \mathbb{C}[x] \right\}$$

- Contains all polynomials we can generate.
- All the polynomials in the ideal are zero on our zero set.
- Tells us about the number of solutions.
- Contains an infinite number of polynomials.
- If the number of solutions is finite there is a monovariate polynomial in this set.

Polynomial Division and rests

- For univariate division is well defined
- For multivariate division the order of monomials is a central issue.

$$\frac{\mathrm{y}^2+\mathrm{x}}{\mathrm{x}+\mathrm{y}} = 1 + \frac{\mathrm{y}^2-\mathrm{y}}{\mathrm{x}+\mathrm{y}} = \mathrm{y} + \frac{-\mathrm{xy}+\mathrm{x}}{\mathrm{x}+\mathrm{y}}$$

Is $\mathrm{x} > \mathrm{y}^2$?

Monomial Orders

Polynomial division requires that the monomials have an order. There exist several monomial orders

- Lexicographic order (phone-book order), very intuitive, gives long computations. Sort by power of x, then by power of y.
- Graded Lexicographic order. Sort first by total degree and then by lexicographic order. More intuitive and quite good for computing Gröbner bases.
- Graded reversed Lexicographic order, very counterintuitive but it is easier to compute a GB.

Gröbner Basis (GB)

- A basis for the ideal.
- Depends on the monomial order chosen.
- Gives the number of solutions.
- There exist algorithms to compute it.
- Gives ways to compute solutions if the number of solutions is finite.
- All leading terms of polynomials of the ideal are divisible by the leading term of some polynomial in the GB.

S-Polynomials

Definition: Given two polynomials f_1 and f_2 with leading terms $x^{\alpha(1)}$ and $x^{\alpha(2)}$. Let x^β be the least common multiple of the leading terms, then

$$S(f_1, f_2) = \frac{x^\beta}{x^{\alpha(1)}} f_1 - \frac{x^\beta}{x^{\alpha(2)}} f_2$$

is the S-polynomial of f_1 and f_2.

- Multiply both polynomials so that their leading terms cancel out.
- Generating S-polynomials is a key step to building a Gröbner basis.

$$S(x^2 + 1, xy + 3) = y(x^2 + 1) - x(xy + 3) = -3x + y$$

Computing a Gröbner basis

There are many algorithms but the main parts are

- Remove linear dependency.
- Remove multiples of known polynomials.
- Generate new S-polynomials.
- Terminate when nothing new can be generated.
- Must use exact arithmetic.
  - Takes time
  - Takes memory

Matrix Representation of polynomials

$$\begin{aligned} 2x^2 + 4xy + y^2 + 8x + y + 1 \\ 2x^2 + 3xy + 2y^2 + 4x + 4y + 4 \end{aligned}$$

can be represented as

$$\begin{bmatrix} 2 & 4 & 1 & 8 & 1 & 1 \\ 2 & 3 & 2 & 4 & 4 & 4 \end{bmatrix} \begin{bmatrix} x^2 \\ xy \\ y^2 \\ x \\ y \\ 1 \end{bmatrix}$$

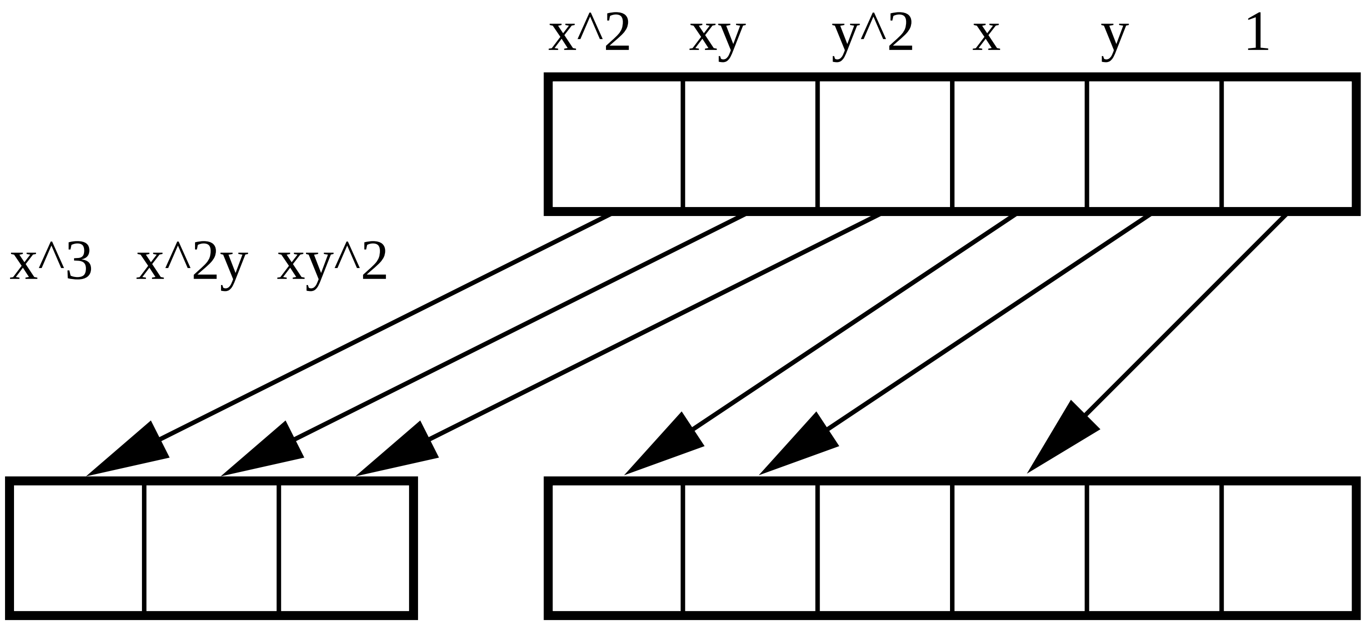
Multiplication with a monomial

- Multiplication is a shift operation

$$(4x + 5y + 6)x = 4x^2 + 5xy + 6x$$

$$\begin{bmatrix} 0 & 0 & 0 & 4 & 5 & 6 \end{bmatrix} \begin{bmatrix} x^2 \\ xy \\ y^2 \\ x \\ y \\ 1 \end{bmatrix} \longrightarrow \begin{bmatrix} 4 & 5 & 0 & 6 & 0 & 0 \end{bmatrix} \begin{bmatrix} x^2 \\ xy \\ y^2 \\ x \\ y \\ 1 \end{bmatrix}$$

Multiplication with a monomial



- Multiplication is a linear operator.

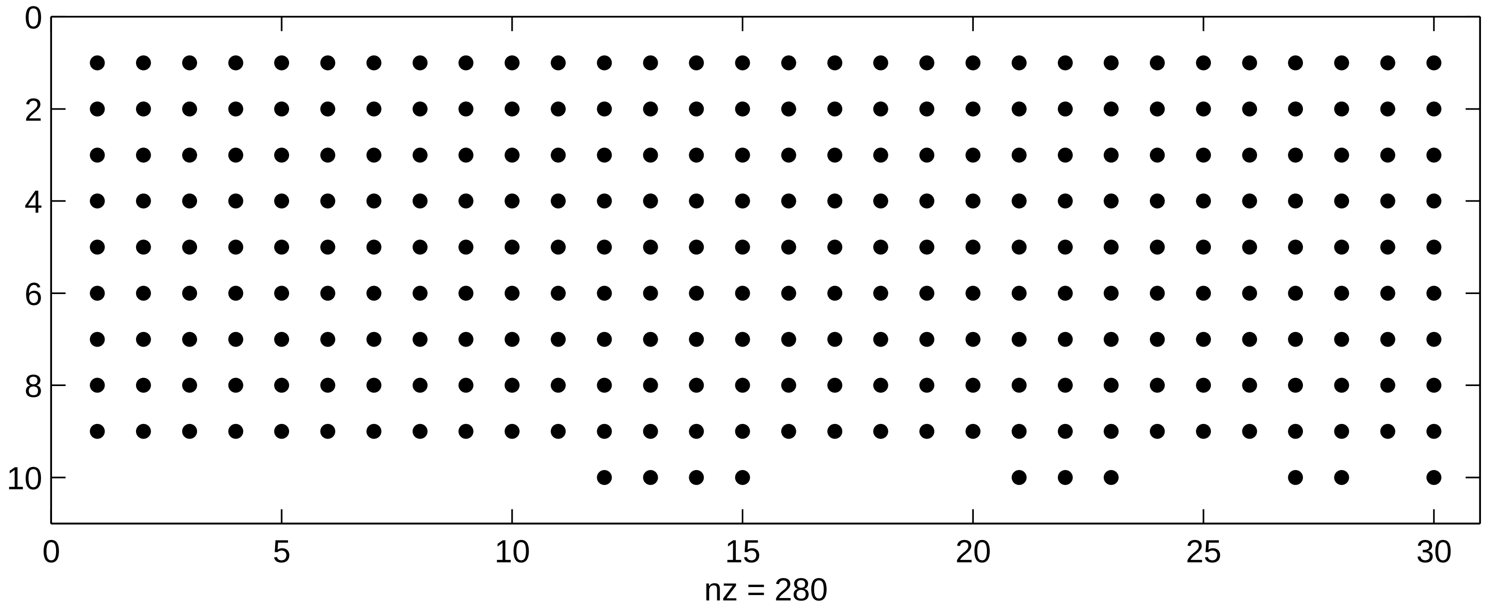
Matrix Based GB Computations

- Add lots of polynomials every time
- Use linear elimination to remove dependencies

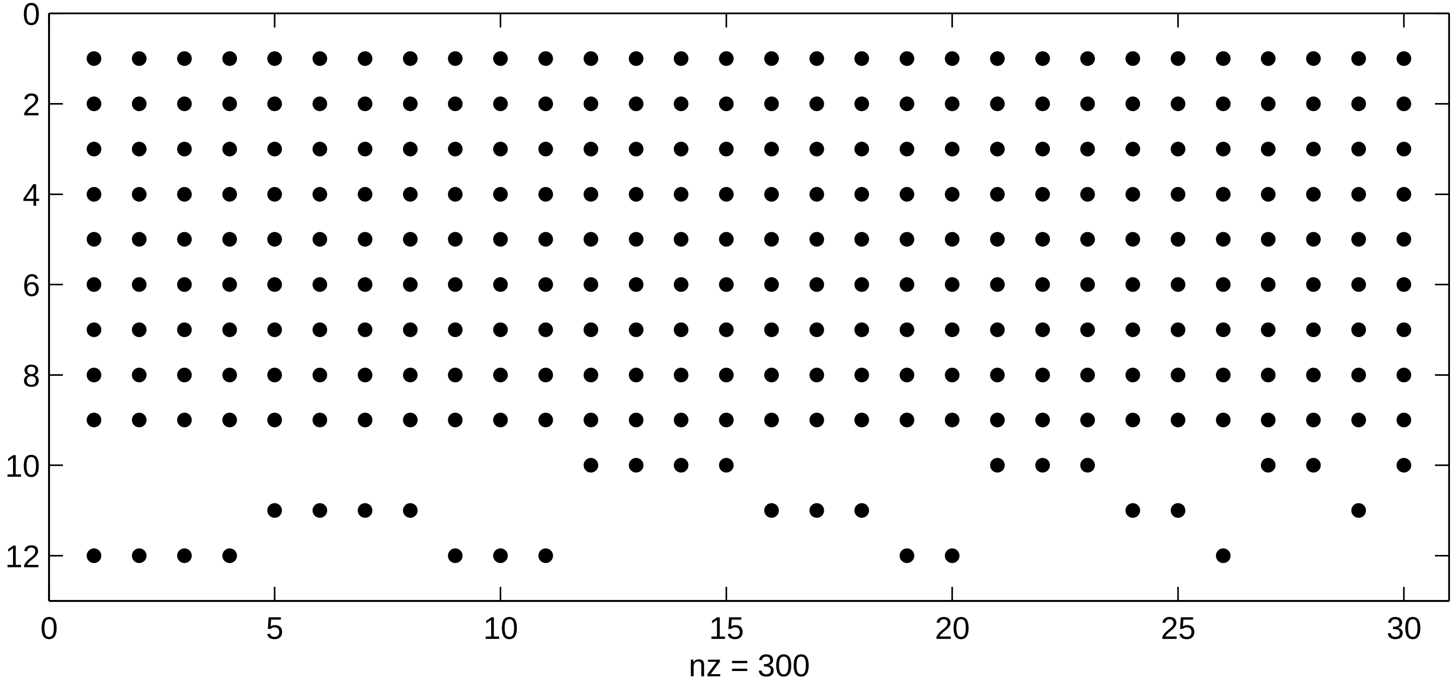
Faugere (F4), Rouillier . . .

- Example: six points and two cameras, unknown focal length.

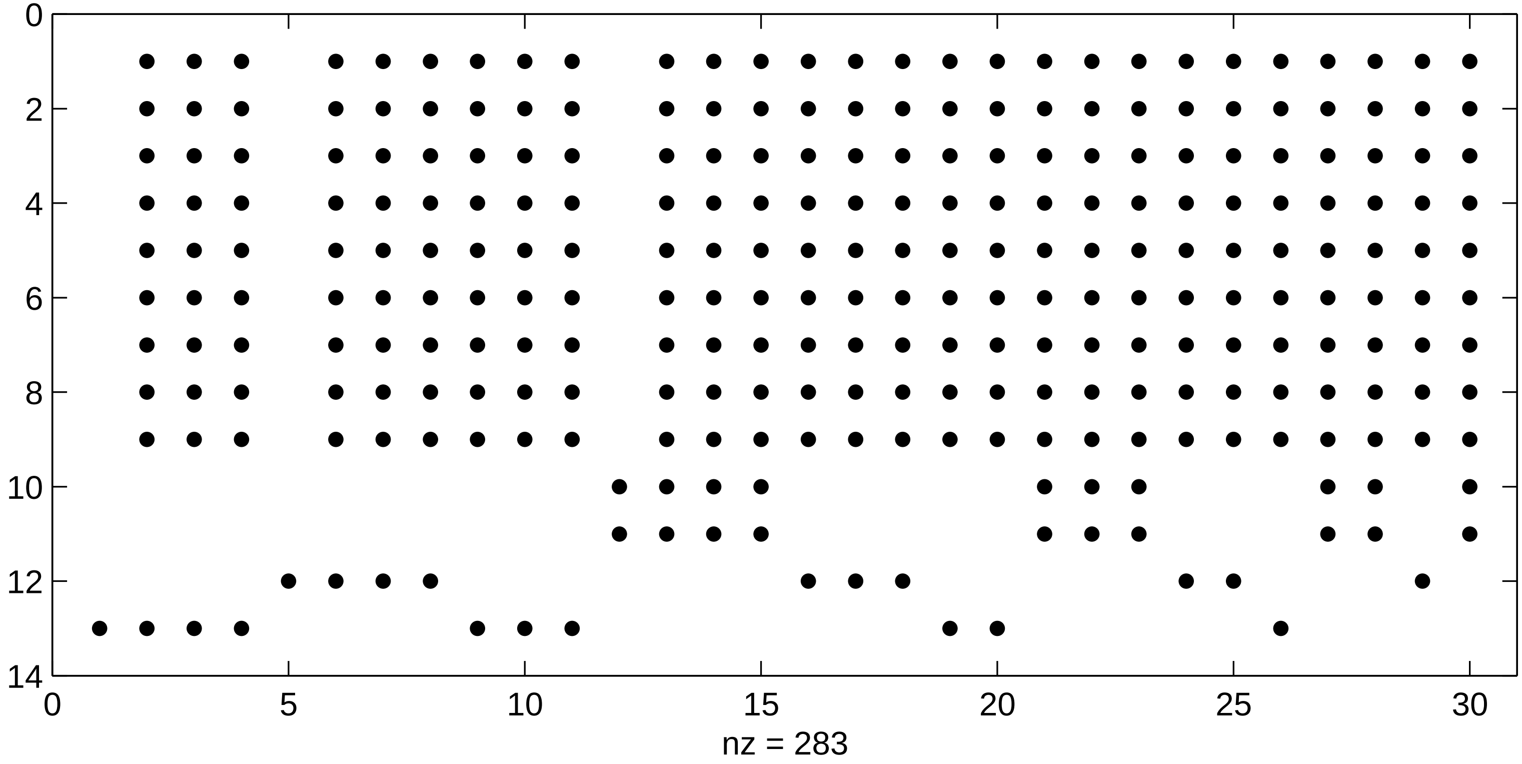
Computational Step



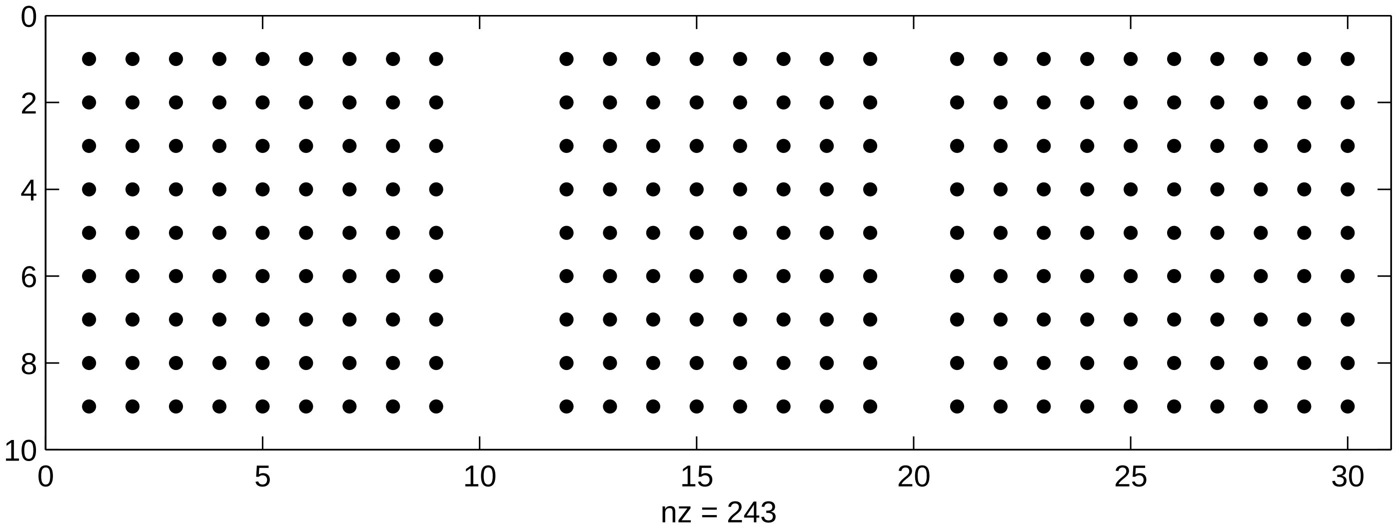
Computational Step



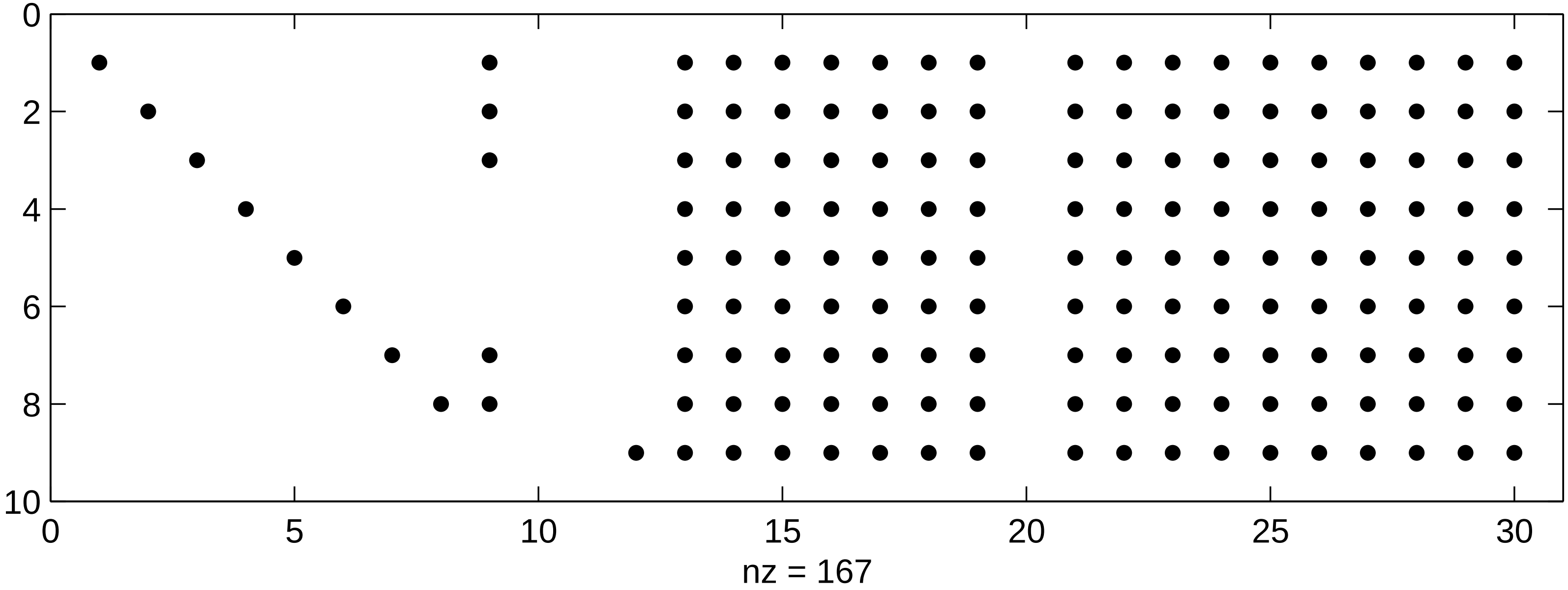
Computational Step



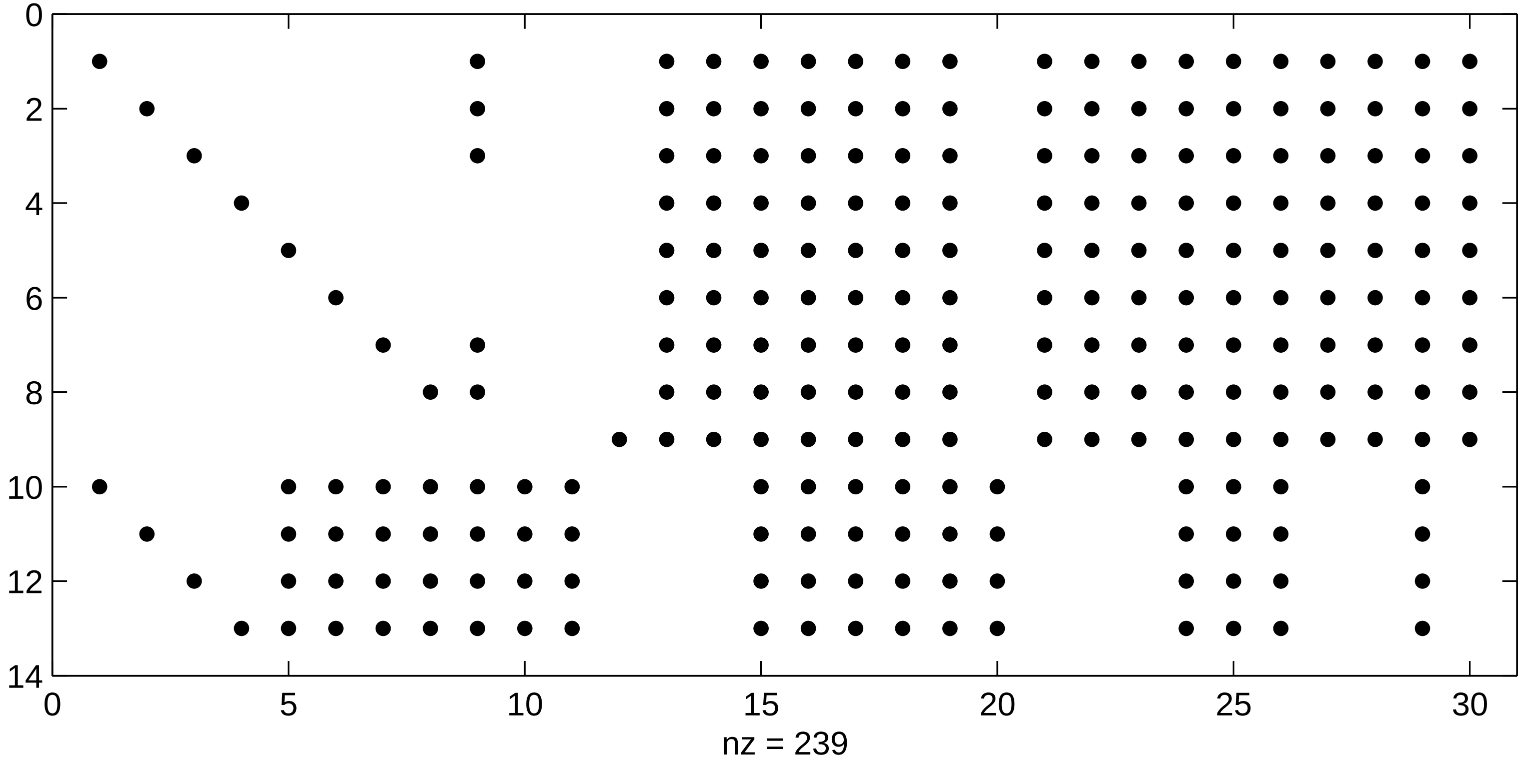
Computational Step



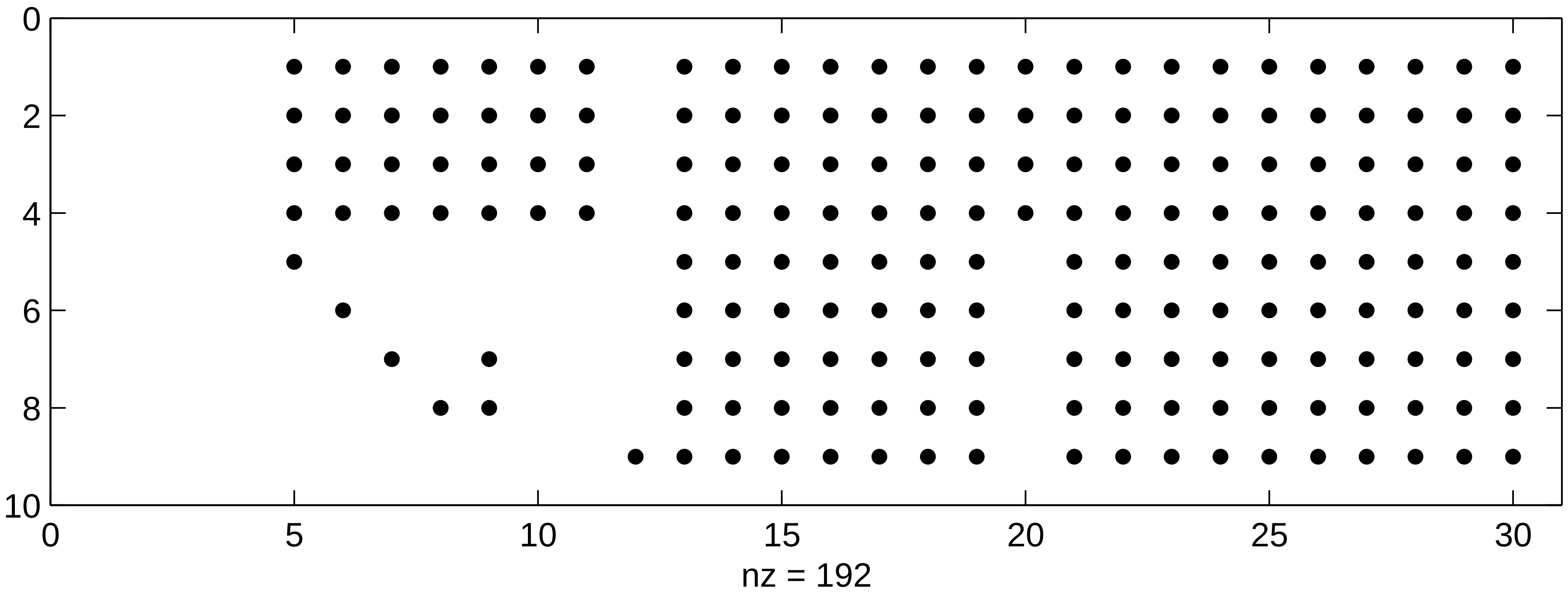
Computational Step



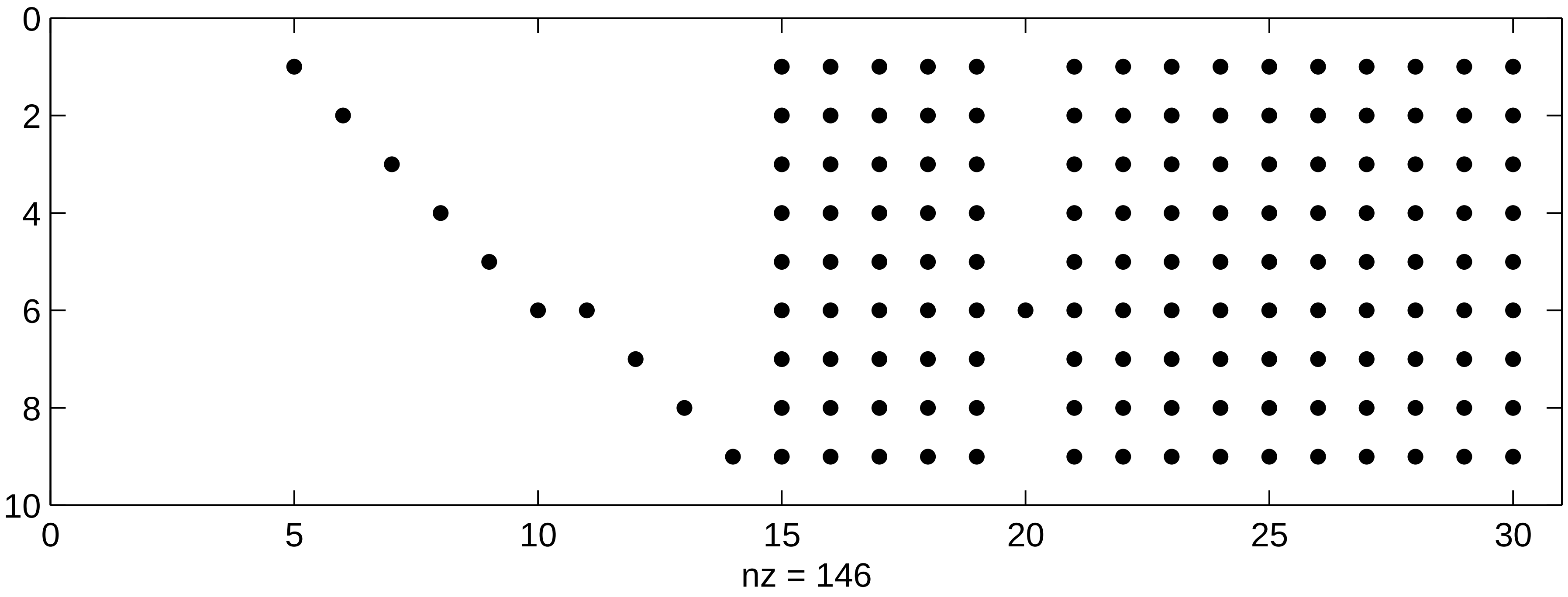
Computational Step



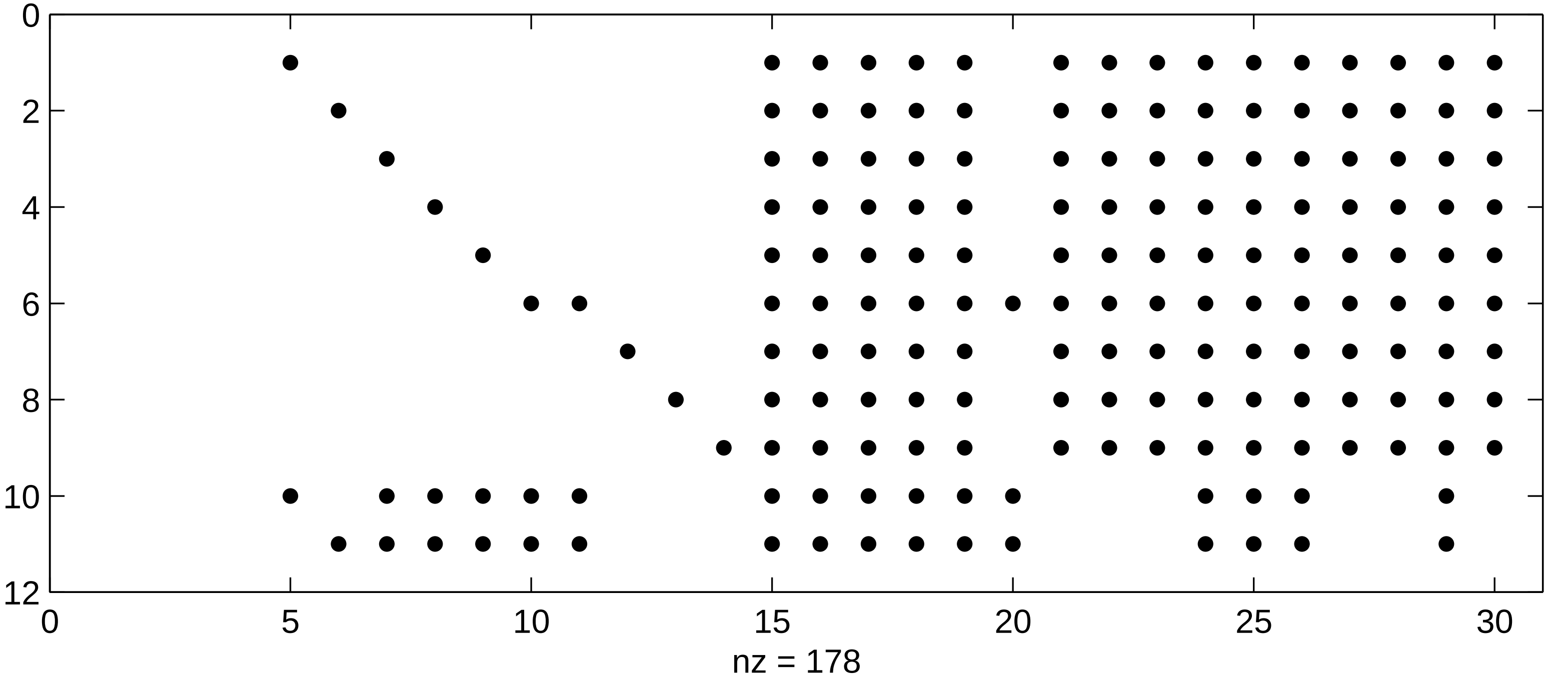
Computational Step



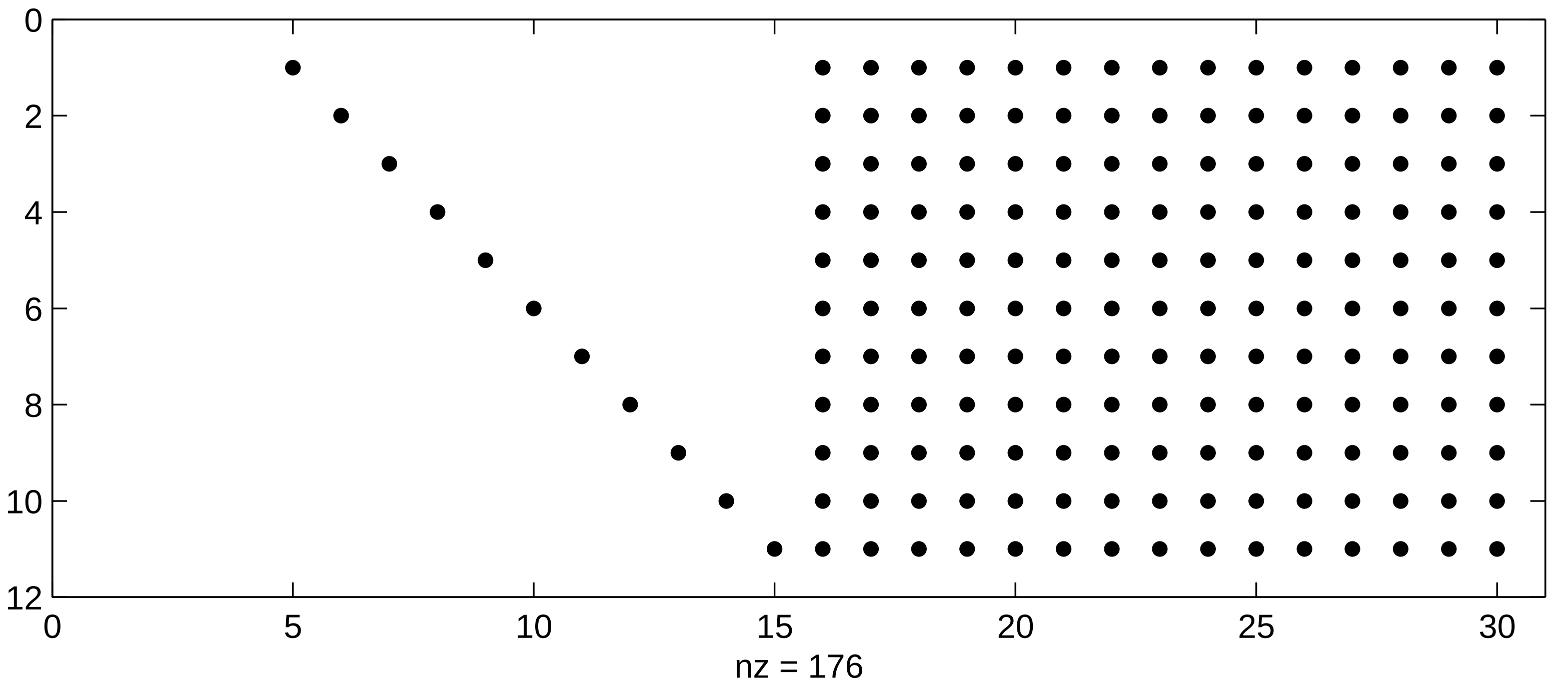
Computational Step



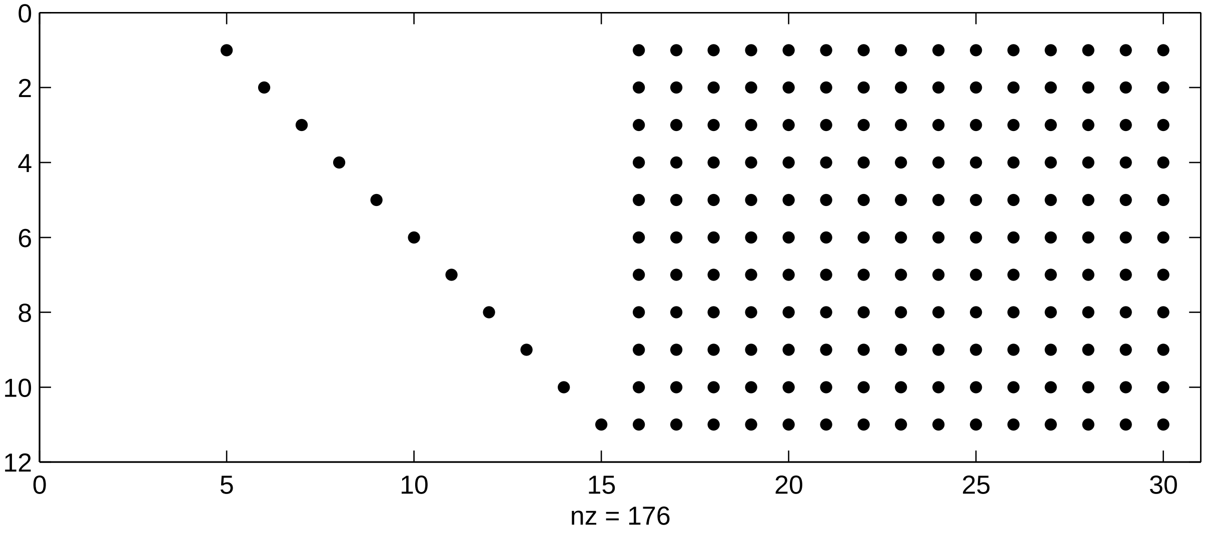
Computational Step

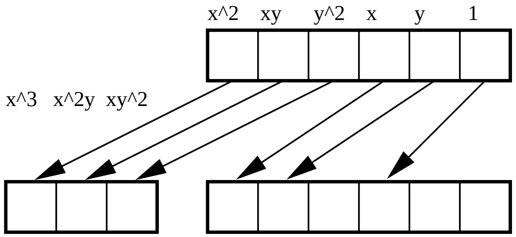


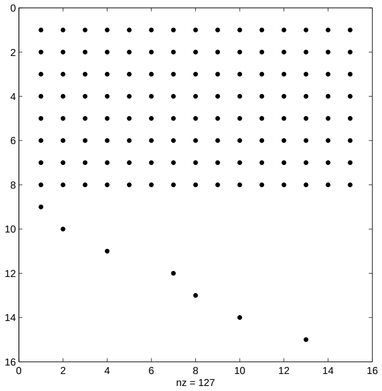
Computational Step



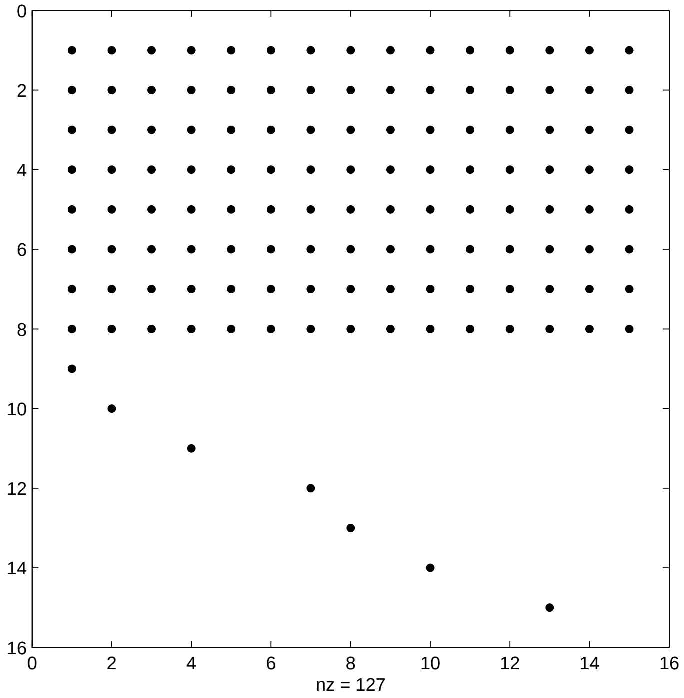
Extracting the Action Matrix







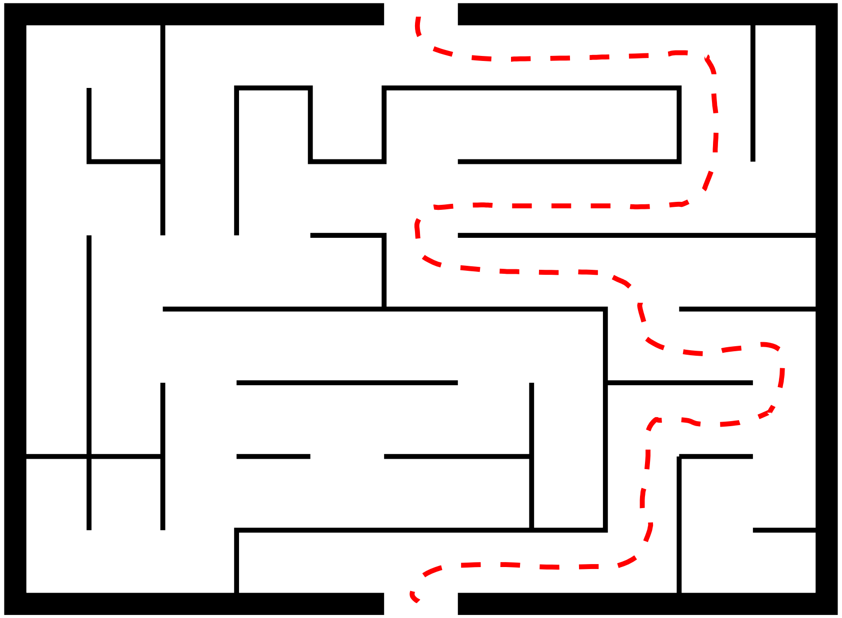
The Action Matrix



- The eigenvectors of this system gives the 15 solutions

The Gröbner Trace

- The trace is always the same in a class of problems.
- Coefficients which become 0 are the walls of a labyrinth.



A finite Field : $\mathbb{Z}_5$

The integers modulo a prime p is a finite field. (Modulo is the rest with division by p.)

- Closed under $+$ and $\cdot$
- Additive and multiplicative inverses

Addition table

	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	5
2	2	3	4	5	6
3	3	4	5	6	0
4	4	5	6	0	1

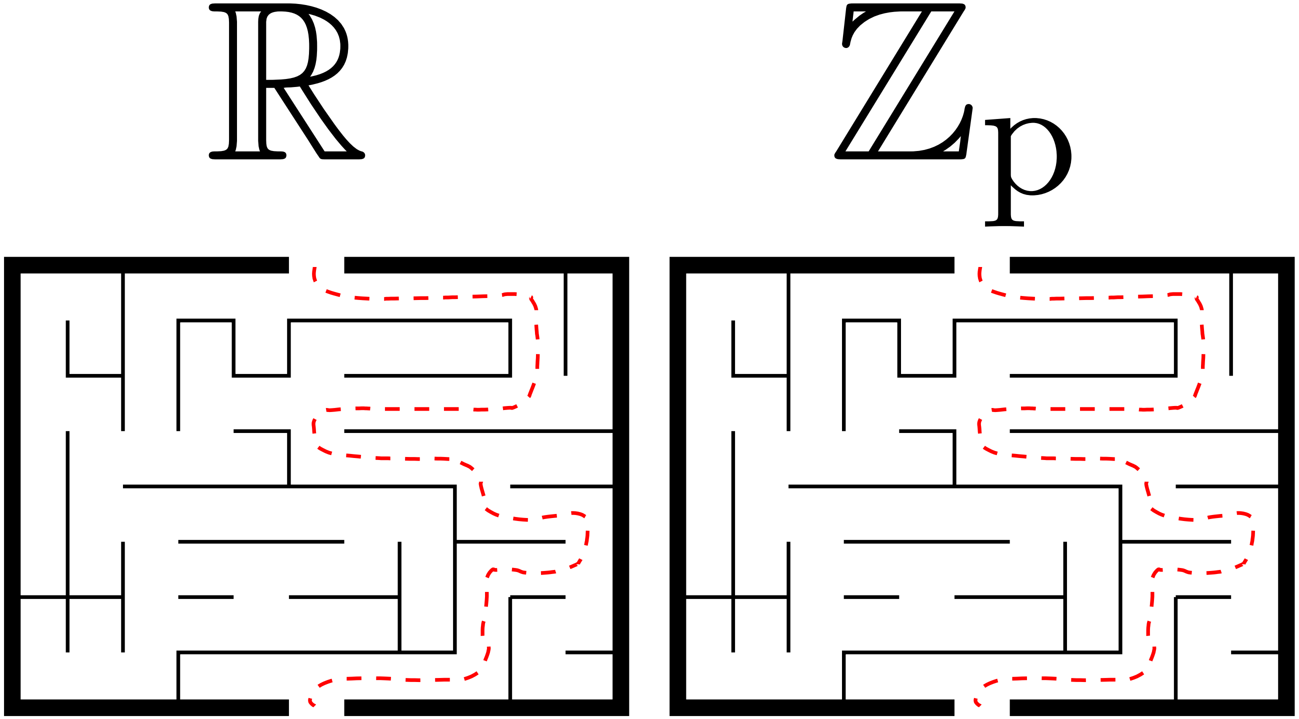
Multiplication table

	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

Why use a finite field?

- Exact arithmetic
  - No problem determining if an element is 0
- Constant memory size for all elements of the field
  - It takes more time to run out of memory
- Gives an excuse to brush of some memories of elementary number theory

Fields are usually similar



- $p = 30097$ is a nice prime.
- Using $\mathbb{Z}_p$ makes it easy to compute the trace.

The Quotient Ideal

- $\mathbb{C}[x]/I$, that is the equivalence classes modulo the ideal I.
  1. Take a polynomial in $\mathbb{C}[x]$
  2. Keep computing rests with polynomials in I until this is no longer possible.
  3. The rest is in $\mathbb{C}[x]/I$
- Example $\mathbb{C}[x]/(x^2 - 1)$
  - 2-dimensional space
  - A basis is 1 and x

The action matrix

Consider what happens to the basis polynomials of the quotient ideal $\mathbb{C}[x]/I$ under multiplication by a polynomial f. This is a linear operator A_f.

- Example: Consider $\mathbb{C}[x]/(x^2 - 1)$ with $f(x) = x$. Then $1 \to x$ and $x \to 1$.

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

- Note the eigenvalues! 1 and -1.

Solution Procedure

1. Pose Problem
2. Count variables and constraints
3. Find GB-trace
4. Compute GB
5. Extract the action matrix
6. Solve Eigenproblem

Helpful tools

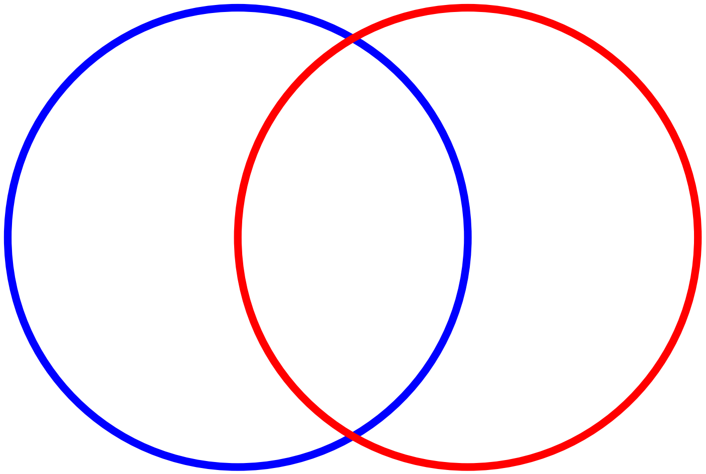
- Macaulay 2,
  `http://www.math.uiuc.edu/Macaulay2/`
  Daniel R. Grayson and Michael E. Stillman, provided under GPL.
- perl
- matlab

Probably all the tools can be substituted for similar tools.

Finding Solutions

- Find all solutions to

$$\begin{cases} x^2 + y^2 & = 1 \\ (x-1)^2 + y^2 & = 1 \end{cases}$$



Howto build an analogous problem 1

```
R = ZZ/30097[x,y];
I = ideal( {x^2+y^2-4 , (x-1)^2+y^2-4 });
dim I , degree I
```

- We get `0,2`

Looking at the trace

```
gbTrace 3
I = ideal( {x^2+y^2-4 , (x-1)^2+y^2-4 });
dim I , degree I
```

Gives

```
mmmr

o12 = (0, 2)
```

m	means an S-polynomial which did not reduce to 0.
r	reduced to 0
o	reduced to 0

Looking at the trace

```
gbTrace 3
I = ideal( {x^2+y^2-4 , (x-1)^2+y^2-4 });
dim I , degree I
```

Gives (filtered a little)

```
--- computing pair (. . x2)  ----
.0  gb 0 = x2<0>+y2<0>-4<0>
--- computing pair (. . x2)  ----
  gb 1 = x<0>+15048<0>
--- computing pair (1 0 x 1 x2)  ----
  gb 2 = y2<0>-7528<0>
--- computing pair (2 1 x y2 xy2 marked)  ----

Number of pairs              = 4
Number of gb elements        = 2
```

Implementing this trace

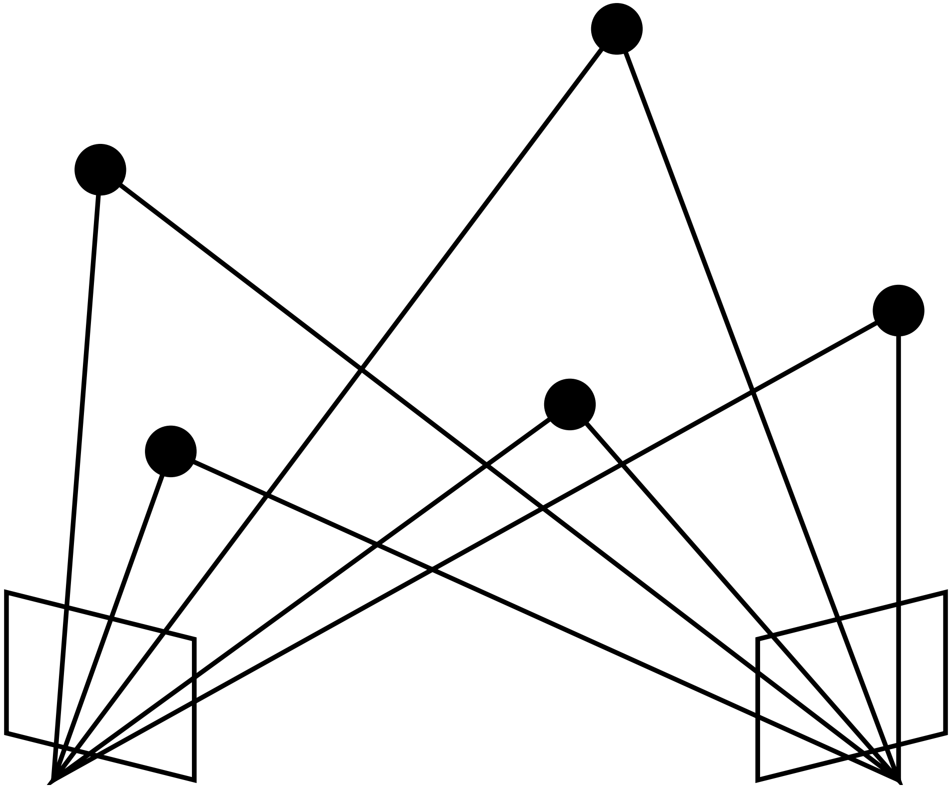
1. Start from $x^2 + y^2 - 1$ and $(x-1)^2 + y^2 - 1$.
2. Linear elimination

$$\begin{bmatrix}1&1&0&-1\\1&1&-2&0\end{bmatrix}\begin{bmatrix}x^2\\y^2\\x\\1\end{bmatrix}\Longrightarrow\begin{bmatrix}1&1&0&-1\\0&0&1&-0.5\end{bmatrix}\begin{bmatrix}x^2\\y^2\\x\\1\end{bmatrix}\Longrightarrow\left\{\begin{array}{rr}x^2+y^2&-1\\x&-0.5\end{array}\right.$$

3. Add $x(x-0.5) = x^2 - 0.5x$ to the basis
4. Linear elimination

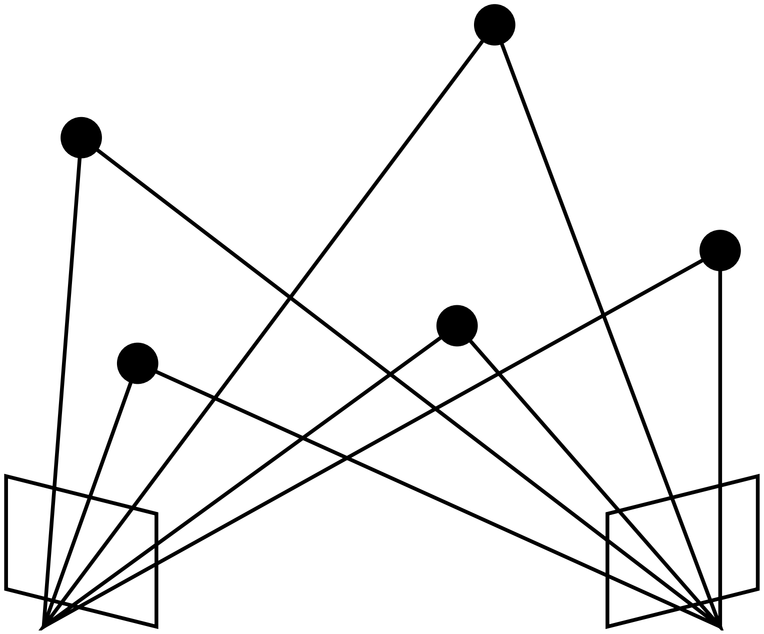
$$\begin{bmatrix}1&1&0&-1\\0&0&1&-0.5\\1&0&-0.5&0\end{bmatrix}\begin{bmatrix}x^2\\y^2\\x\\1\end{bmatrix}\Longrightarrow\begin{bmatrix}1&0&0&0.25\\0&1&0&-0.75\\0&0&1&-0.5\end{bmatrix}\begin{bmatrix}x^2\\y^2\\x\\1\end{bmatrix}\Longrightarrow\left\{\begin{array}{lrrr}x^2&&&+0.25\\&y^2&&-0.75\\&&x&-0.5\end{array}\right.$$

Relative Pose



- Observation Equation $\lambda_{ij} u_{ij} = P_i U_j$.
- Alternative Observation equation $x^\top F x' = 0$.
- Restrictions on P or F.

Finding Solutions



- Five points and two cameras relative pose

$$\begin{cases} \mathrm{E} = \mathrm{E}_0 + \mathrm{x}\mathrm{E}_1 + \mathrm{y}\mathrm{E}_2 + \mathrm{z}\mathrm{E}_3 \\ 2\mathrm{E}\mathrm{E}^\top\mathrm{E} - \mathrm{trace}(\mathrm{E}\mathrm{E}^t)\mathrm{E} = 0 \\ \det(\mathrm{E}) = 0 \end{cases}$$

Howto build an analogous problem 2

Investigating the analogous problem for the 5-point problem.

```
R=ZZ/30097[x_0..x_2];
t={1_R,x_0,x_1,x_2};
E=fold((a,b)->a+b,apply(4,i->t_i*random(R^3,R^3)));
EEt = E*transpose(E);
I = ideal( 2*EEt*E-trace(EEt)*E )+ ideal det E;
gbTrace 3
dim I , degree I
```

Howto build an analogous problem

- Set up the finite field (3 variables)

```
R = ZZ/30097[x_0..x_2];
```

- Declare a list

```
t = {1_R,x_0,x_1,x_2};
```

- Set $E = E_3 + \sum_{i=0}^{2} t_i E_3$ where E_i random.

```
E=fold((a,b)->a+b,apply(4,i->t_i*random(R^3,R^3))
```

- Write down the equations, take the ideal

```
EEt = E*transpose(E);
I = ideal( 2*EEt*E-trace(EEt)*E )+ ideal det E;
```

- Set the amount of information from the computation

```
gbTrace 3
```

- Compute the dimension and degree of the ideal

```
dim I , degree I
```

The output

```
i7 : dim I , degree I
mmmmmmmmmmooooooooooooooo

o7 = (0, 10)
```

There are 10 solutions of dimension 0

More output

Set `gbTrace 100`, rerun computations and filter

```
--- computing pair (. . x_0^3)  ----
.0  gb 0 = x_0^3
     .
     .
--- computing pair (. . x_0^3)  ----
  gb 9 = x_2^3
--- computing pair (9 8 x_1 x_2 x_1x_2^3)  ----
     .
     .
.21--- computing pair (1 0 x_0 x_1 x_0^3x_1)  ----
Number of gb elements       = 10
Number of pairs computed    = 25
```

We see that no new S-polynomials where generated.

Implementing this trace

1. Set the 10 equations into a system with coefficients ordered in GrLex order.

```
MONS = [3 2 2 1 1 1 0 0 0 0 2 1 1 0 0 0 1 0 0 0;
        0 1 0 2 1 0 3 2 1 0 0 1 0 2 1 0 0 1 0 0;
        0 0 1 0 1 2 0 1 2 3 0 0 1 0 1 2 0 0 1 0];
```

2. Perform Gaussian elimination

$$\begin{bmatrix} \mathrm{I} & \mathrm{M} \end{bmatrix}$$

3. We now have a Gröbner basis

To get a complete solver we have to extract the action matrix.

`http://www.vis.uky.edu/~stewe/FIVEPOINT/`

The filter I used

```
rm tt
cat class25_new.m2 | M2 > tt
cat tt|perl -p -e 's/gb/##gb/;s/comput/##comput/;s/\<0\>.*//'
      |egrep "##" | perl -p -e 's/##//'  | egrep -v marked
```

(one line-break inserted where there should be none)

Conclusions from M2

- Study the number of solutions
- Study if the trace looks possible
- This is an iterative process

The Problem Formulation is Very Important

- Get as many equations as possible
- Keep the degree as low as possible
- Keep the number of variables as low as possible

Tricks

- Use fundamental/essential matrix if possible.
- Use determinants to remove variables.
- ...

6 point, unknown f

- Stewénius, Nistér, Kahl and Shaffalitzky, CVPR2005.
- Using the trace constraint and the rank constraint building a GB is very straight forward.
- 15 solutions

6 point, unknown f

$$F = F0 + l_1F_1 + l_2F_2$$

```
KK = ZZ / 30097;
R = KK[l_1,l_2,p];
F0 = random(R^3,R^3);
F1 = random(R^3,R^3);
F2 = random(R^3,R^3);
F = F0 + l_1*F1 + l_2*F2;
```

6 point, unknown f

$$E = K^\top F K$$

```
P = matrix {{1,0,0},{0,1,0},{0,0,p}};
tr = X -> det( X_{0}^{0} +X_{1}^{1}+X_{2}^{2});
EQ1=2*F*P*transpose(F)*P*F-tr(P*F*P*transpose(F))*F;
EQ2=det( F);
I = ideal EQ1 + ideal EQ2;
gbTrace 3;
dim I , degree I
```

Do not drop equations!

- Variable focal length (10 equations)

```
mmmmmmmmmmmmmmmmmmmmmmmmooooooooo
oooooooorooooroorrrrrororoor
15 solutions
```

- Variable focal length (2 equations)

```
mmmmmmmmmmmmmmmmommmmmmooooooooo
oooorooooroorrrrr
27 solutions
```

Generalized Camera in M2

- Generalized Camera (good problem formulation)

```
mmmmmmmmmmooomoomommooooooooommmmmmmm
mmmmmmommoommmooooooooooooooooooomooo
ooooooooooororrrororoo
```

64 solutions

- Generalized Camera (bad problem formulation)

```
mmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmm
mmmmmmmmmmmommmommmmommmmmmmoommmmmmm
mmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmmm
            ...... (total 168 lines)
rrrrrorrrrrrrrorrrrrrrrrrrrorrrrrrrrrr
```

64 solutions

Implementing a Trace

We have a formulation where the trace looks ok

1. Write a matrix version in `M2`.
2. Build the equations in real numbers
3. Port the matrix version to `Matlab` or `C`

A Matrix Version in M2

- M2 was not meant for matrix operations
- Need to hack our own Gaussian elimination
  - It can be done using the Gröbner engine
  - It is also possible to hack it using special commands. Please email me for source.

Extracting the equations

Two alternatives

- Symbolically expand the equations and identify the coefficients (slower)
- Build a vector based multiplication, much like convolution. (faster)

It is strongly recommended to use some symbolic toolbox for this. For more complex expressions the expansion can be very long!

Porting a solver

1. Put the original equations into a matrix
2. Derive the multiplication matrices
3. Work carefully. check against a problem where the ground truth is known.

Setting up the monomials in GrevLex order

```
function MONS = build_MONS_3var(n)
MONS =[];
for i1=0:n
   for i2=0:n
      if i1+i2 <= n
         MONS = [MONS [i1;i2]];
      end
   end
end
[dummy,I]=sortrows(  [sum(MONS);  -MONS(end:-1:1,:) ]
MONS = MONS(:,I(end:-1:1));
```

Build a multiplication matrix

```
function M = build_mult_matrix( MONS_from,MONS_to,mul
% MONS_* matrices where the columns represent monomia
% mult is a column matrix representing a monomial
%
mult = mult(:)';
M =[];
for i=1:size(MONS_from,2)
   p = MONS_from(:,i)';
   M=[M ismember(  MONS_to', p+mult , 'rows' ) ];
end
M = M';
```

Extract the Action Matrix

For the 5-point the action matrix is read by the following commands.

```
M = -A([1 2 3 5 6 8], :);
M(7,1) = 1;
M(8,2) = 1;
M(9,4) = 1;
M(10,7) = 1;
```

- Use code to build this code! (does not fit this page)

Action Matrix to Solutions

```
[V,D] = eig(M );
SOLS =   V(7:9,:)./(ones(3,1)*V(10,:));
```

and then back-substitute

```
Evec = EE*[SOLS ; ones(1,10 ) ];
```

planarly restricted generalized camera

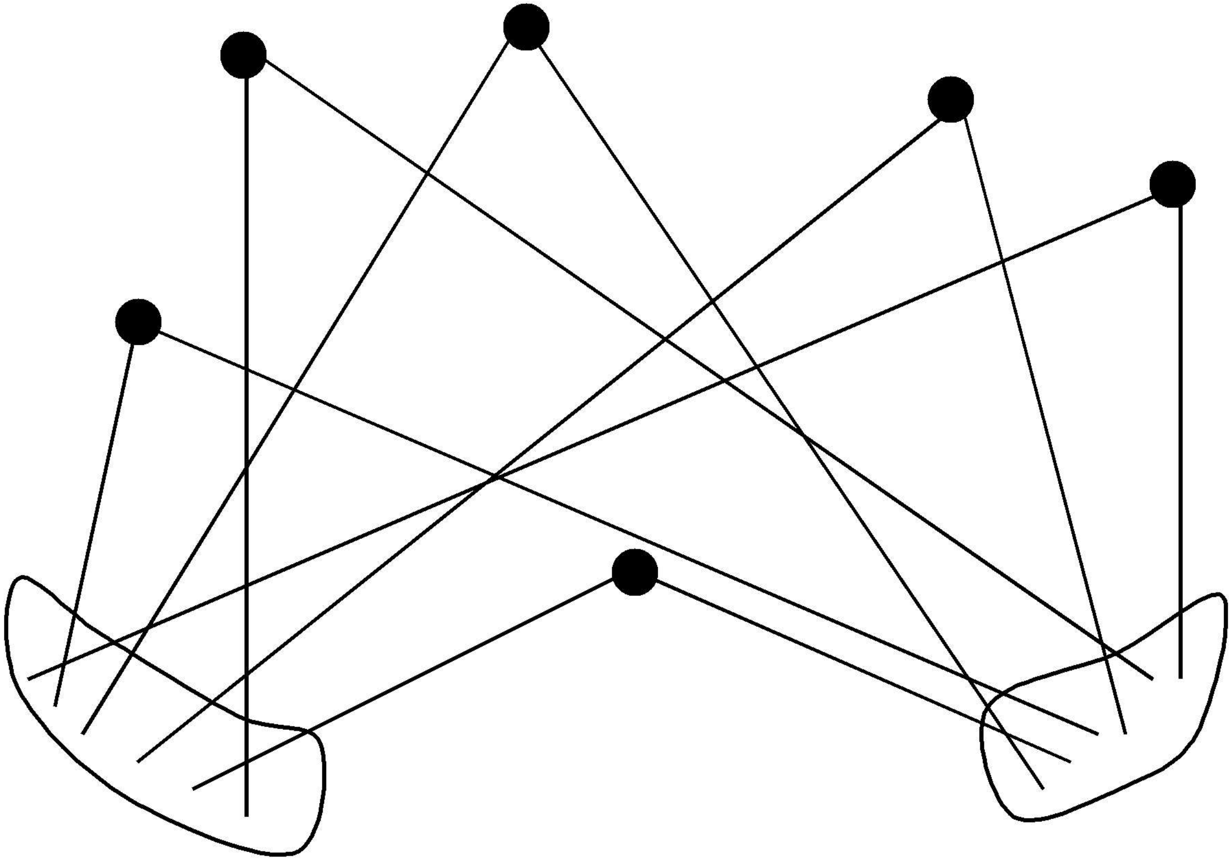
- H. Stewénius and K. Åstrom ECCV2004, before we fully understood how to work with GB.
- There are two easy minimal relative pose problems.
- This case is suitable as an easy exercise to begin with.

Relative Pose generalized camera

- There are 2 minimal cases, 64 solutions and 1320 solutions.
- For the easy problem:
  - Stewénius, Nistér, Oskarsson and Åström at OMNIVIS 2005.
  - Departing from the generalized epipolar constraint.
  - Requires a rather hard-to-find rewrite

The Generalized Camera

- A Generalized Camera maps from points in space to lines in the camera coordinate system.
  $(\mathrm{X}, \mathrm{T}) \rightarrow (\mathrm{q}, \mathrm{q}')$



The Generalized Epipolar Constraint

- $R_1 = I$
- R_2 is the rotation at time i
- t_i is the translation at time i
- (q_i, q_i') is the observation in the camera

$$q_2^\top R_2 q_1' + q_2^\top (R_2 [t_1]_\times - [t_2]_\times R_2) q_1 + q_1^\top R_2^\top q_2' = 0$$

- This gives a polynomial

Parameterizing with depth i

Let $\mathrm{U}_1 = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T$ define the coordinate system. Then

$$\mathrm{t}_\mathrm{j} = \mathrm{q}'_{\mathrm{kj}} \times \mathrm{q}_{\mathrm{kj}} + \lambda_{\mathrm{kj}} \mathrm{q}_{\mathrm{kj}} \quad \mathrm{j} = 1, 2, \qquad \underbrace{\mathrm{F}^\mathrm{k}(\mathrm{v})}_{5\times 3} \begin{bmatrix} \lambda_{\mathrm{k}1} \\ \lambda_{\mathrm{k}2} \\ 1 \end{bmatrix} = 0$$

- $\mathrm{F}^\mathrm{k}(\mathrm{v})$ is quadratic in v
- rank condition gives 6th order equations on v

$$\begin{bmatrix} [2] & [2] & [2] \\ [2] & [2] & [2] \\ [2] & [2] & [2] \\ [2] & [2] & [2] \\ [2] & [2] & [2] \end{bmatrix}$$

Using a second asymmetry

- Choosing $U_1 = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T$ gives 10 linearly independent equations.
- Choosing $U_2 = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T$ gives 10 linearly independent equations.

Together this gives 14 equations

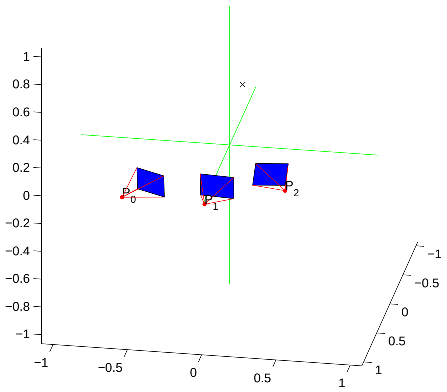
Constructing a GB

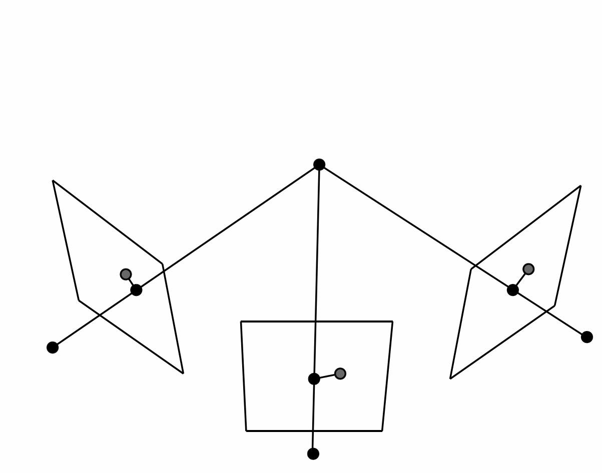
1. Add the polynomials $\mathrm{v_j f_i}$ there are now 56 polynomials
2. Compute the missing polynomial as an S-polynomial

We know have a GrevLex Gröbner-basis.

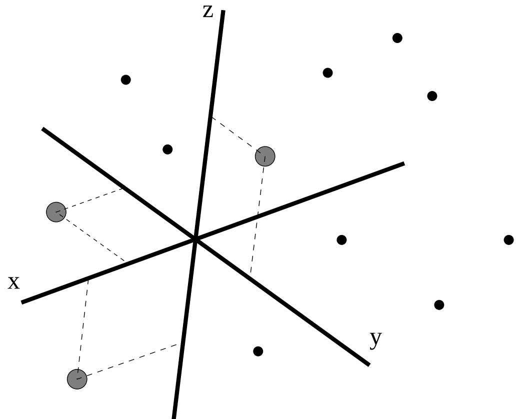
- As there are 64 inner points to the Gröbner basis the action matrix is 64 × 64. This gives a total of 64 solutions.

3-view triangulation, L_2-norm





- There are 47 stationary points to the target function
- Have to remove unwanted zeros (saturation)



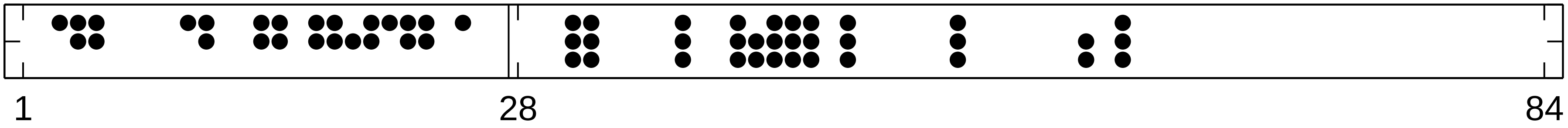
Removing Zeros

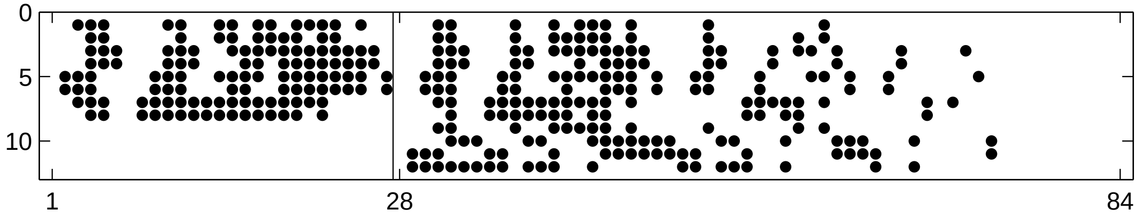
We want to remove the set defined by $f(x) = 0$ from the solutions. Two different possibilities:

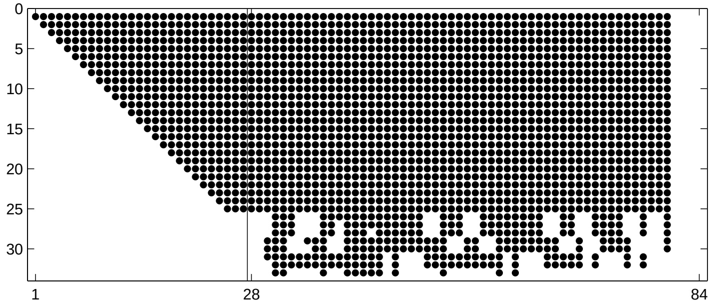
- Rabinovichs trick. Add the synthetic variable s and the polynomial $1 - sf(x)$.
- Divide out the unwanted solutions.
    - Use special monomial order. (Elimination order)

Saturation

- Saturation, three rounds







- Data flow

I_0 Partial Saturation I_1 Partial Saturation I_2 Partial Saturation I_3

Some help

- Construct test to verify results halfway …
- Do not write by hand what the computer can write for you
- Feel free to browse in my poorly commented code archive
- The slides will be available soon

  `http://www.vis.uky.edu/~stewe/`

End

The slides and some code will be available on:

`http://www.vis.uky.edu/~dnister/`

`http://www.vis.uky.edu/~stewe/`