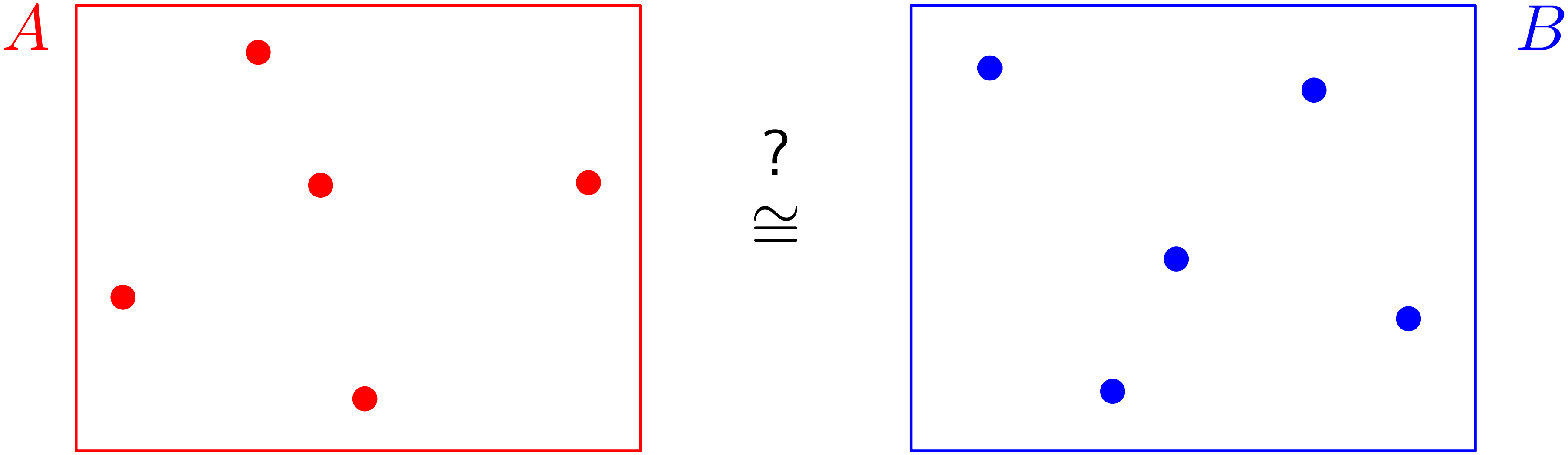
# Congruence Testing in 4 Dimensions

Günter Rote

joint work with Heuna Kim

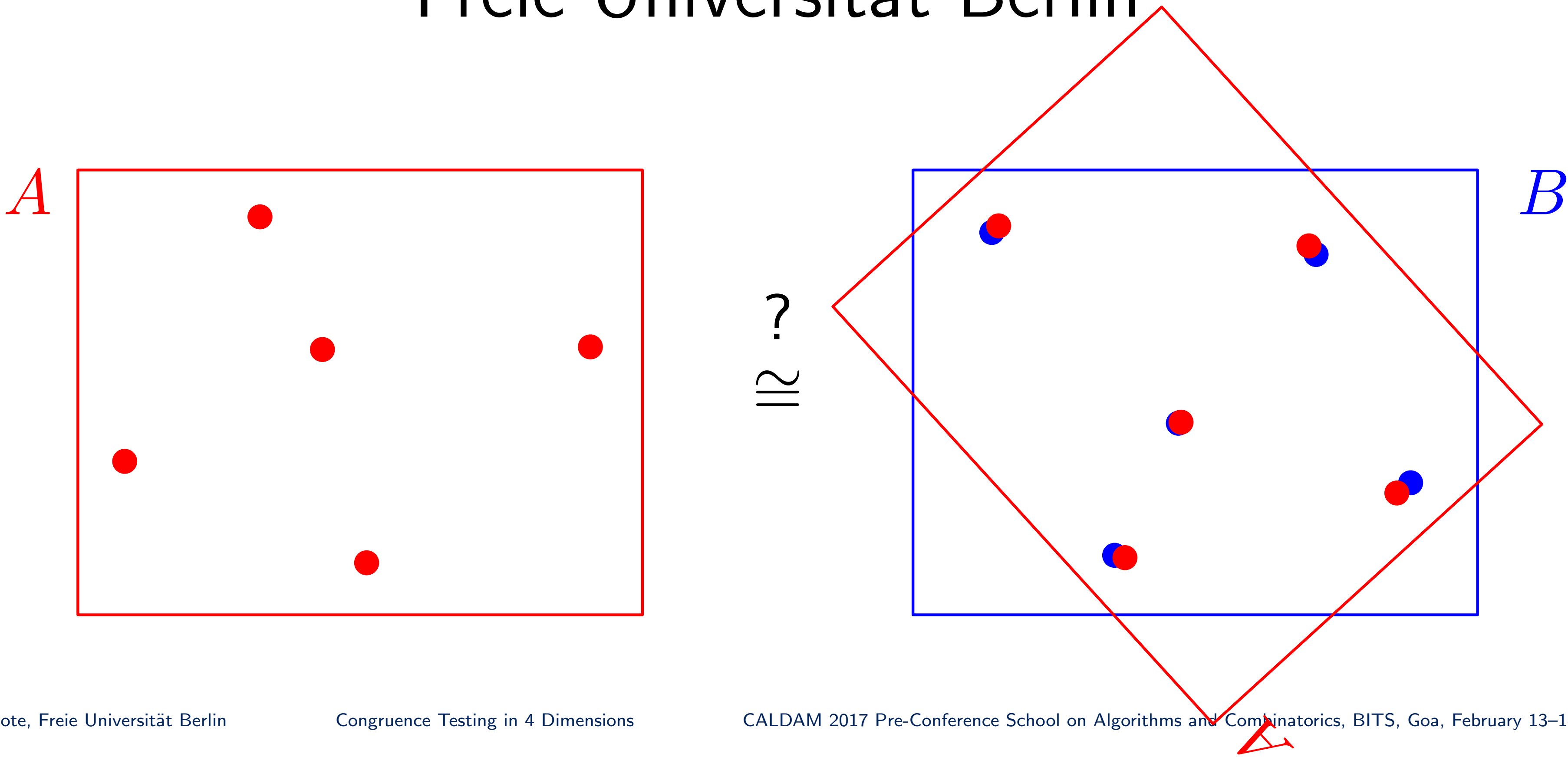
Freie Universität Berlin



# Congruence Testing in 4 Dimensions

Günter Rote
joint work with Heuna Kim

Freie Universität Berlin



# Overview

- 1 dimension
- 2 dimensions
- 3 dimensions
- 4 dimensions
- $d$ dimensions

# Overview

- 1 dimension
- 2 dimensions
- 3 dimensions
- 4 dimensions

$O(n \log n)$ time

NEW, joint work with Heuna Kim

- $d$ dimensions $O(n^{\lceil d/3 \rceil} \log n)$ time [Brass and Knauer 2002]

$O(n^{\lfloor d/2 \rfloor /2} \log n)$ time Monte Carlo[Akutsu 1998/Matoušek]

# Overview

- 1 dimension
- 2 dimensions
- 3 dimensions
- 4 dimensions

} $O(n \log n)$ time

NEW, joint work with Heuna Kim

- $d$ dimensions $O(n^{\lceil d/3 \rceil} \log n)$ time [Brass and Knauer 2002]

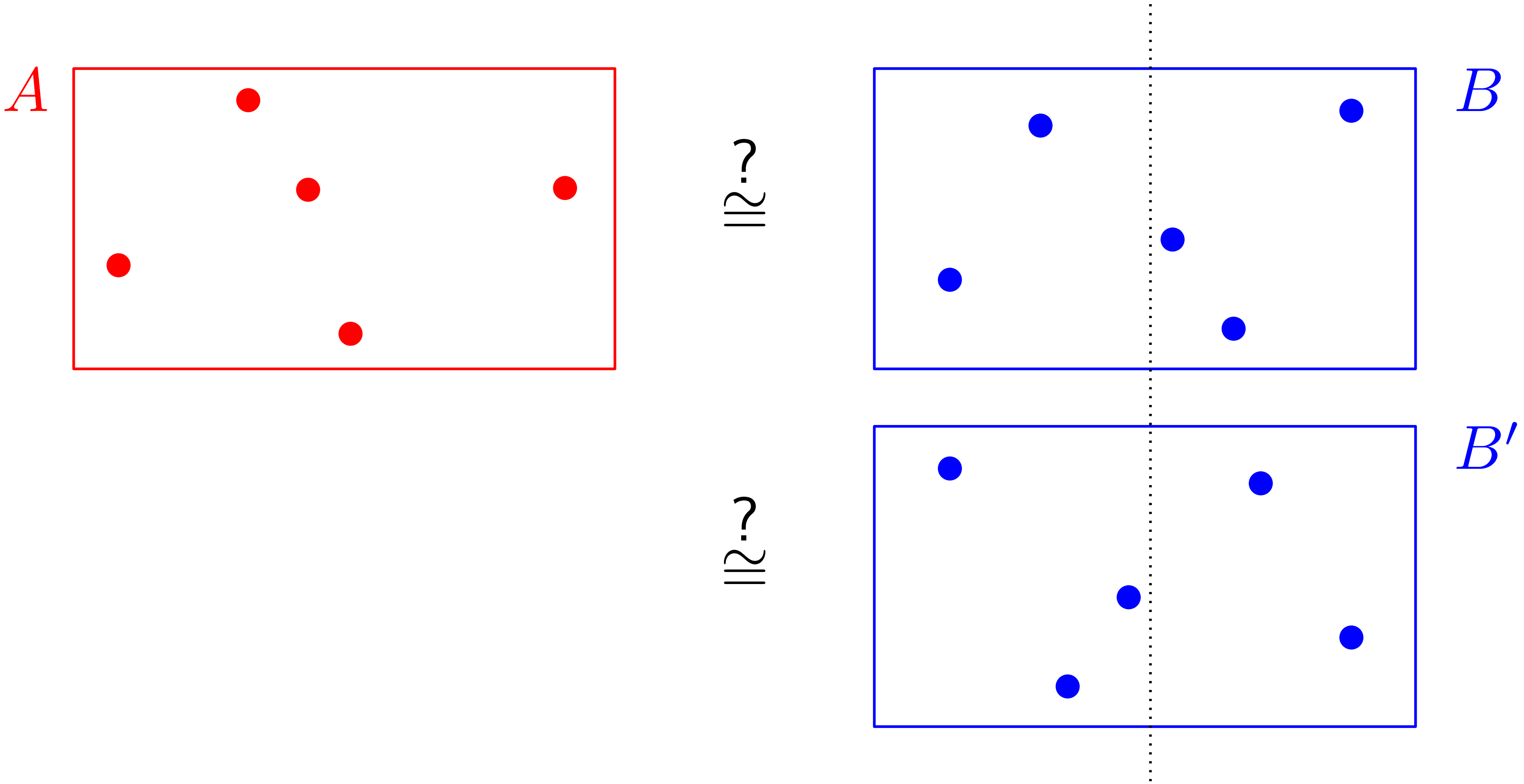
  $O(n^{\lfloor d/2 \rfloor /2} \log n)$ time Monte Carlo[Akutsu 1998/Matoušek]

- Problem statement and variations
- Dimension reduction as in [Alt, Mehlhorn, Wagener, Welzl]
- Atkinson's reduction (pruning/condensation)
- (Planar) graph isomorphism
- Hopf fibrations
- Plücker coordinates
- Coxeter groups

# Rotation or Rotation+Reflection?

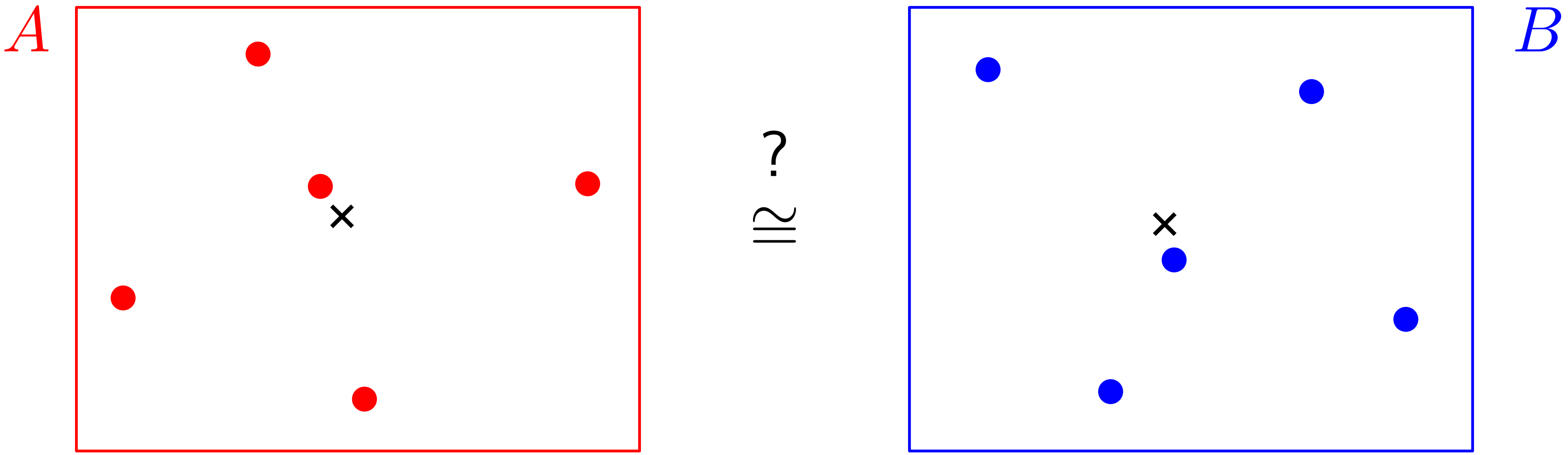
We only need to consider *proper* congruence (orientation-preserving congruence, of determinant $+1$).

If mirror-congruence is also desired, repeat the test twice, for $B$ and its mirror image $B'$.



# Congruence = Rotation + Translation

Translation is easy to determine:
The centroid of $A$ must coincide with the centroid of $B$.



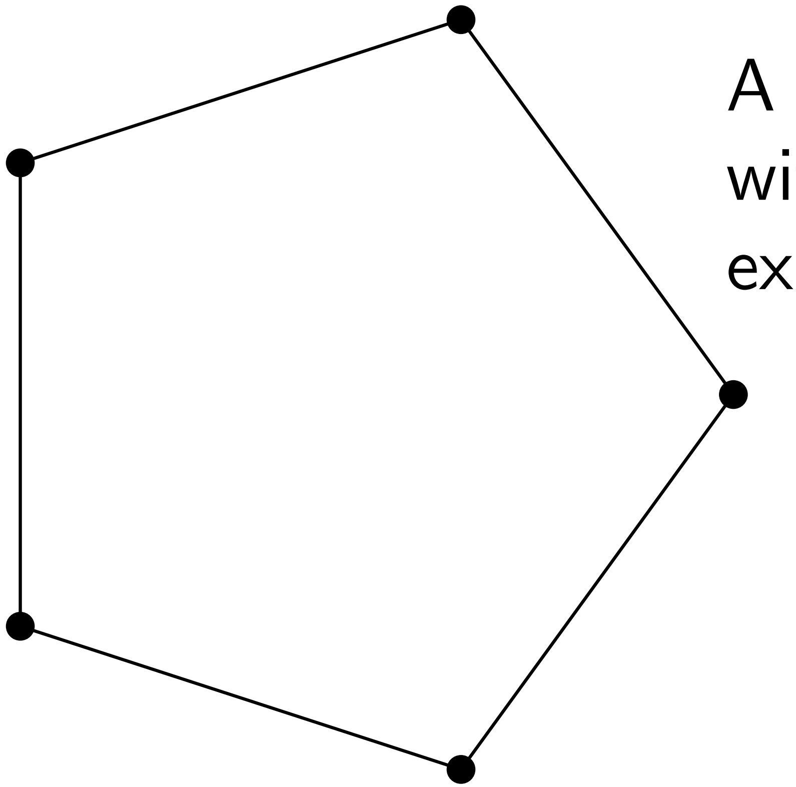
→ from now on: All point sets are centered at the origin $0$:

$$\sum_{a\in A} a = \sum_{b\in B} b = 0$$

We need to find a rotation around the origin (orthogonal matrix $T$ with determinant $+1$) which maps $A$ to $B$: $TA = B$

# Exact Arithmetic

The proper setting for this (mathematical) problem requires real numbers as inputs and exact arithmetic.
→ the *Real RAM* model (RAM = random access machine):
One elementary operation with real numbers ($+$, $\div$, $\sqrt{}$, $\sin$) is counted as one step.



A regular 5-gon, 7-gon, 8-gon, ... with rational coordinates does not exist in any dimension.

# Applications

Congruence testing is the basic problem for many pattern matching tasks

- computer vision
- star matching
- brain matching
- . . .

The proper setting for this applied problem requires tolerances, partial matchings, and other extensions.

# Arbitrary Dimension

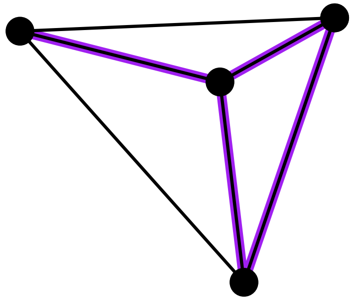
$A, B \subset \mathbb{R}^d$, $|A| = |B| = n$.

We consider the problem for fixed dimension $d$.

When $d$ is unrestricted, the problem is equivalent to graph isomorphism:

$G = (V, E)$, $V = \{1, 2, \ldots, n\}$

$$\mapsto A = \underbrace{\{e_1, \ldots, e_n\}}_{\text{regular simplex}} \cup \{ \tfrac{e_i + e_j}{2} \mid ij \in E \} \subset \mathbb{R}^n$$



$$e_i = (0, \ldots, 0, 1, 0, \ldots, 0)$$

CONJECTURE:

Congruence can be tested in $O(n \log n)$ time for every fixed dimension $d$.

Current best bound: $O(n^{\lceil d/3 \rceil} \log n)$ time

# Arbitrary Dimension

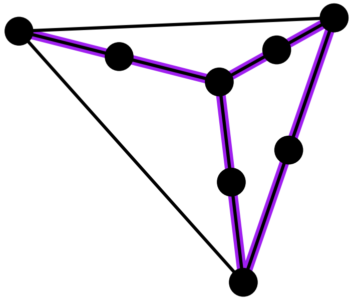
$A, B \subset \mathbb{R}^d$, $|A| = |B| = n$.

We consider the problem for fixed dimension $d$.

When $d$ is unrestricted, the problem is equivalent to graph isomorphism:

$G = (V, E)$, $V = \{1, 2, \ldots, n\}$

$$\mapsto A = \underbrace{\{e_1, \ldots, e_n\}}_{\text{regular simplex}} \cup \{ \tfrac{e_i + e_j}{2} \mid ij \in E \} \subset \mathbb{R}^n$$



$$e_i = (0, \ldots, 0, 1, 0, \ldots, 0)$$

CONJECTURE:

Congruence can be tested in $O(n \log n)$ time for every fixed dimension $d$.

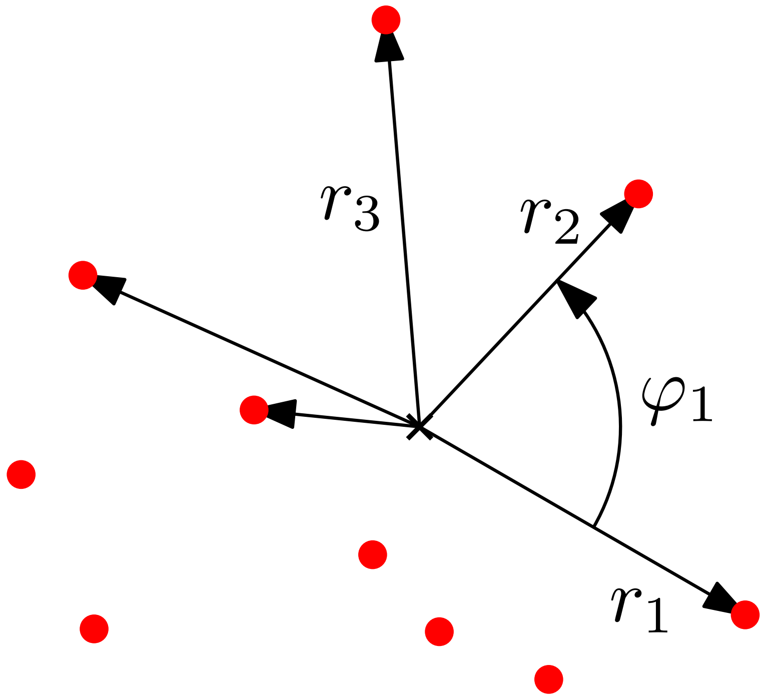
Current best bound: $O(n^{\lceil d/3 \rceil} \log n)$ time

# Two dimensions

Can be done by string matching. [ Manacher 1976 ]

Sort points around the origin.

Encode alternating sequence of distances $r_i$ and angles $\varphi_i$.



$$(r_1, \varphi_1, r_2, \varphi_2, \ldots, r_n, \varphi_n)$$

Check whether the corresponding sequence of $B$ is a cyclic shift.

$\to O(n \log n) + O(n)$ time.

# Two dimensions

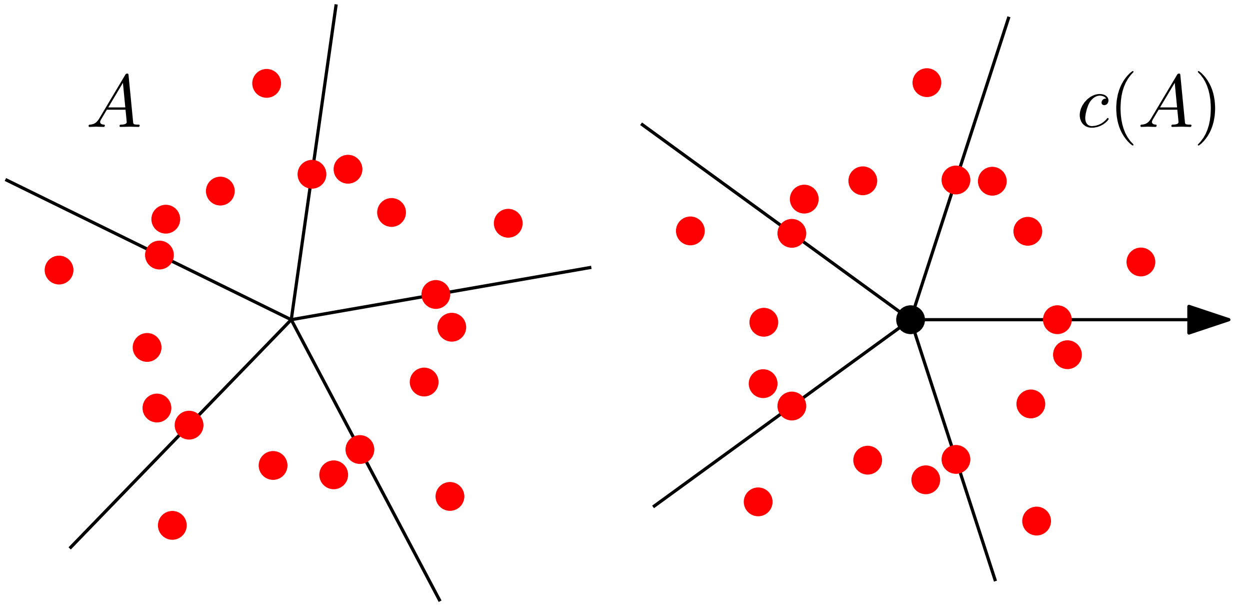
Can be done by string matching. [ Manacher 1976 ]

Sort points around the origin.

Encode alternating sequence of distances $r_i$ and angles $\varphi_i$.

Even more can be done:

CANONICAL DIRECTIONS



The *canonical set* $c(A)$: [Akutsu 1992]

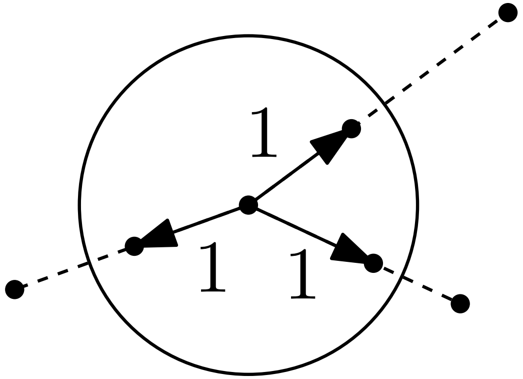
$$A \cong B \iff c(A) = c(B)$$

$\rightarrow$ searching in a database

# Three dimensions

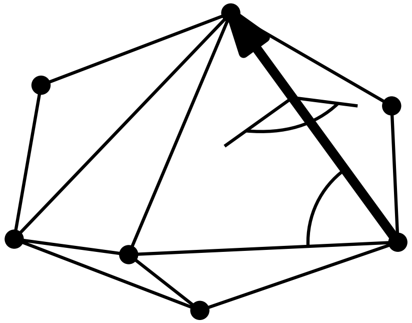
[ Sugihara 1984; Alt, Mehlhorn, Wagener, Welzl 1988 ]

Project points to the unit sphere, and keep distances as *labels*.



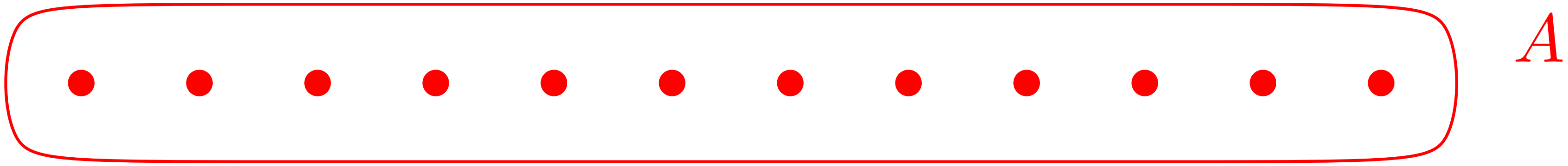
Compute the convex hulls $P(A)$ and $P(B)$, in $O(n \log n)$ time.

Check isomorphism between the corresponding LABELED planar graphs.
Vertex labels: from the radial projection
Edge labels: dihedral angles and face angles.

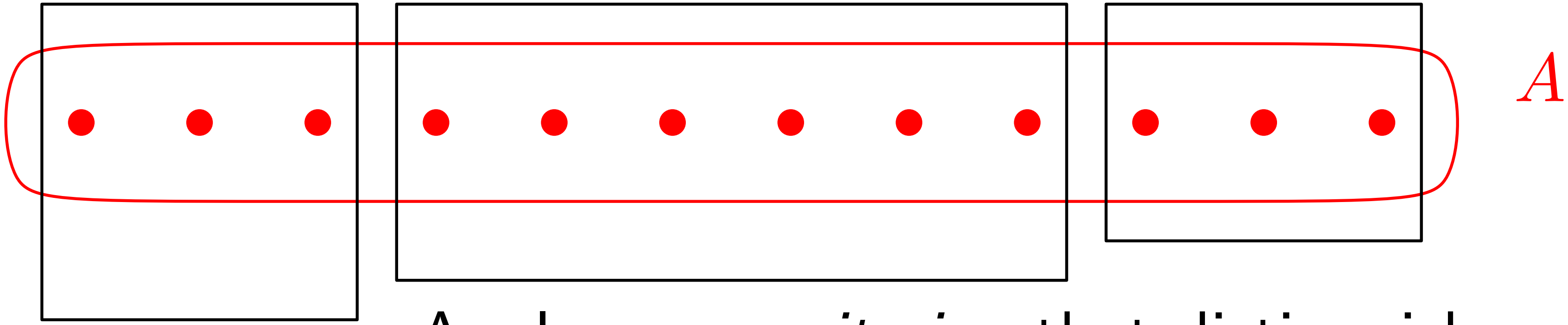


In $O(n)$ time, [ Hopcroft and Wong 1974 ]
or in $O(n \log n)$ time. [ Hopcroft and Tarjan 1973 ]

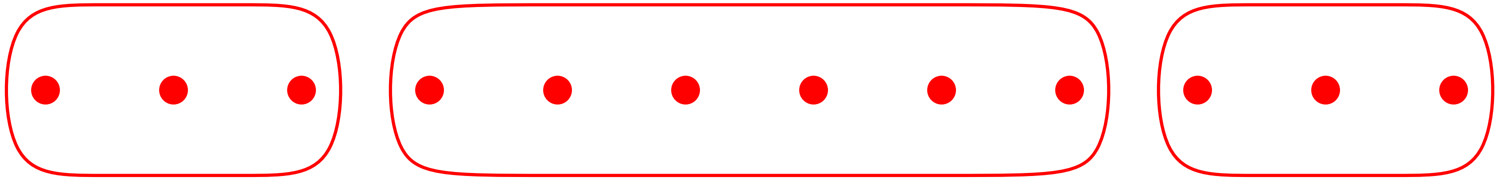
# Pruning/Condensing



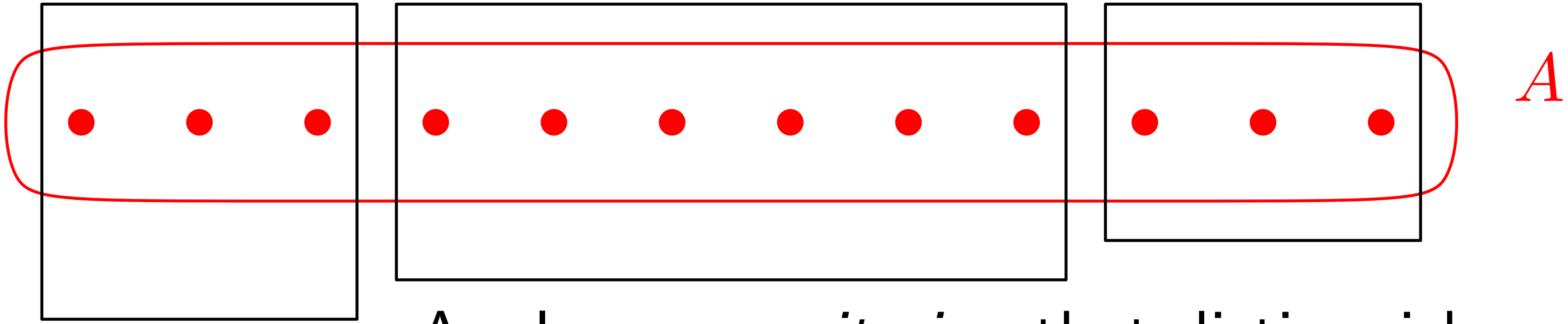
# Pruning/Condensing



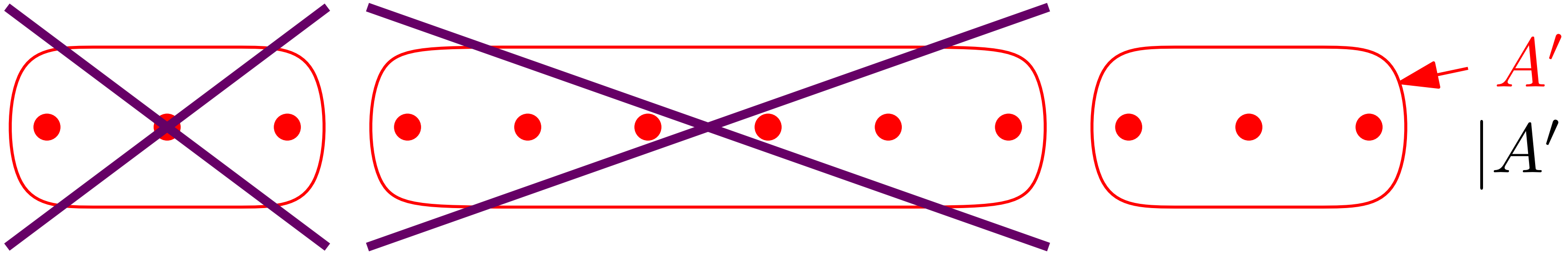
Apply some *criterion* that distinguishes points (distance from the center, number of closest neighbors, . . . )



# Pruning/Condensing



Apply some *criterion* that distinguishes points
(distance from the center,
number of closest neighbors, . . . )

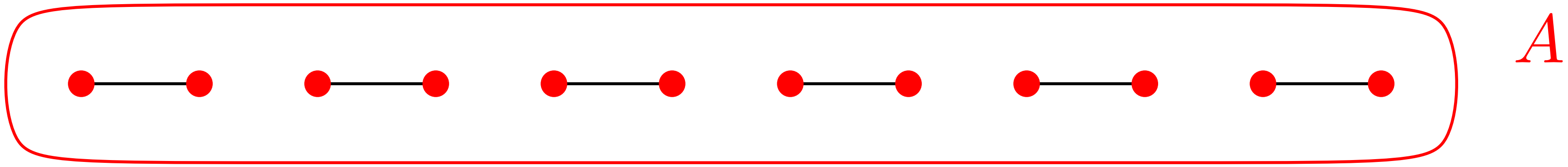


$|A'| \leq \frac{|A|}{2}$

Throw away all but the smallest
resulting class,
and repeat.

Simultaneously apply this procedure to $B$.
$A'$ and $B'$ may have *more* congruences!

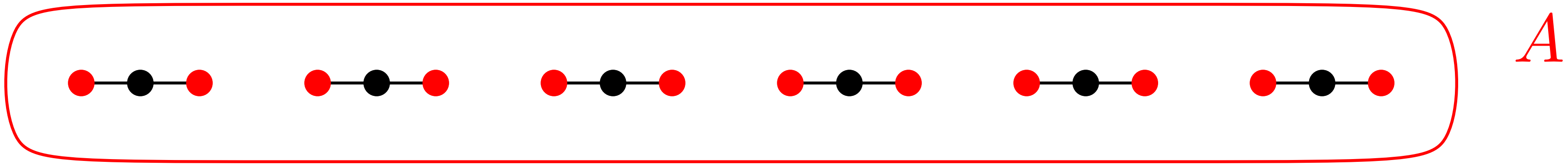
# Pruning/Condensing



Make some *construction*
(midpoints of closest-pair edges, . . . )

Simultaneously apply this procedure to $B$.
$A'$ and $B'$ may have *more* congruences!

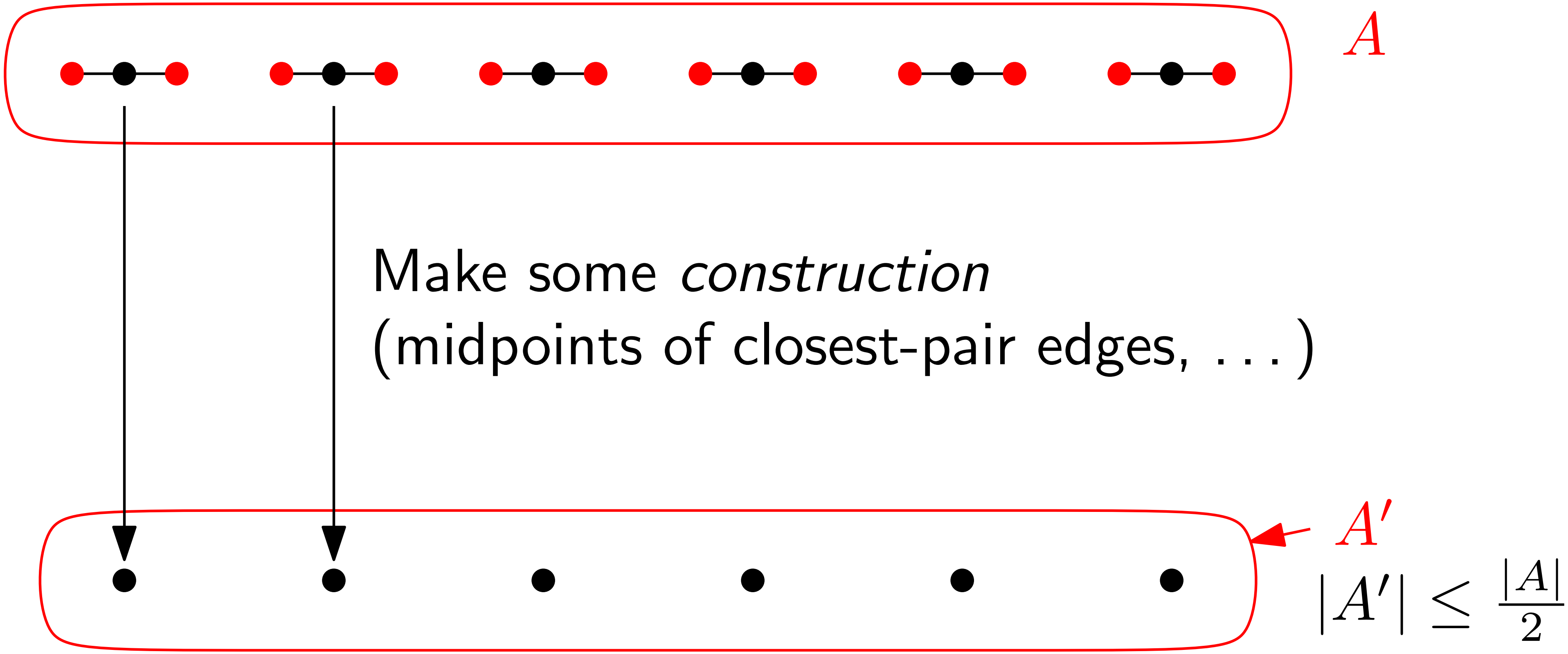
# Pruning/Condensing



Make some *construction*
(midpoints of closest-pair edges, . . . )

Simultaneously apply this procedure to $B$.
$A'$ and $B'$ may have *more* congruences!

# Pruning/Condensing



Simultaneously apply this procedure to $B$.

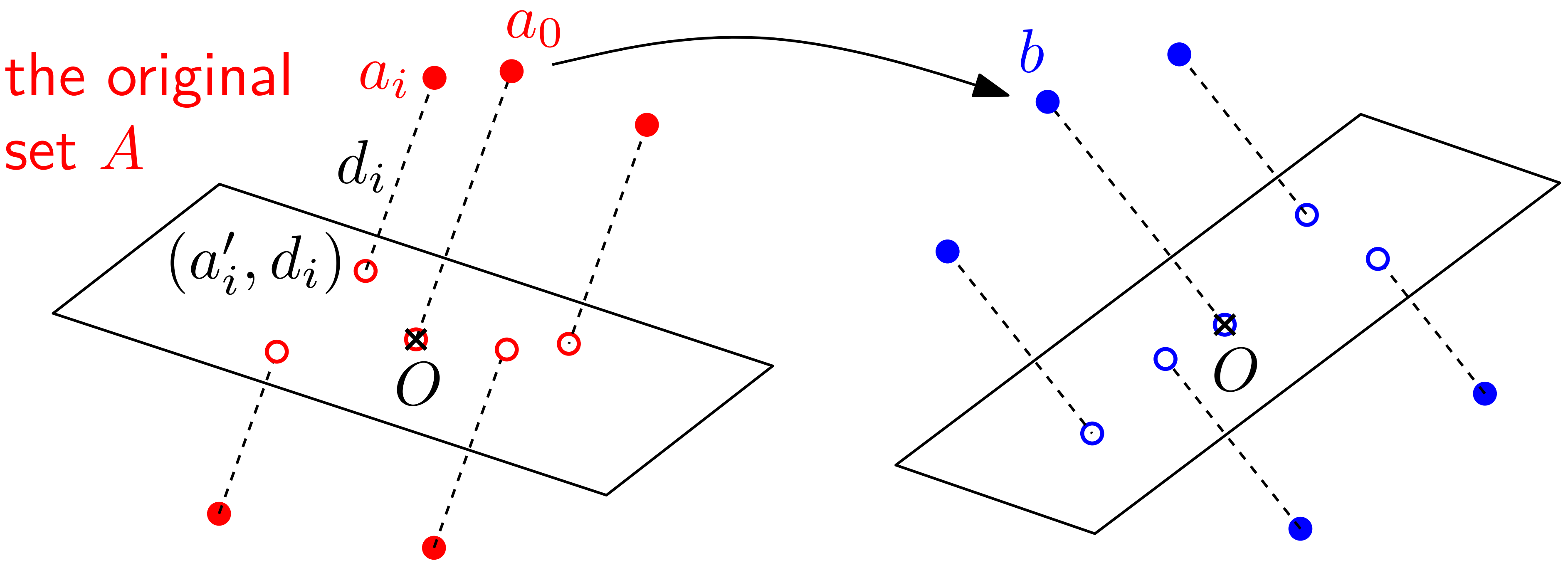
$A'$ and $B'$ may have *more* congruences!

# Dimension Reduction

As soon as $|A'| = |B'| = k$ is small:

Choose a point $a_0 \in A'$ and try all $k$ possibilities of mapping it to a point $b \in B'$.

$$\boxed{\text{Fixing } a_0 \mapsto b \text{ reduces the dimension by one.}}$$



Project perpendicular to $Oa_0$ and label projected points $a'_i$ with the signed projection distance $d_i$ as $(a'_i, d_i)$.

→ 2-dimensional congruence for LABELLED point sets

# Dimension Reduction

As soon as $|A'| = |B'| = k$ is small:
Choose a point $a_0 \in A'$ and try all $k$ possibilities of mapping it to a point $b \in B'$.

$\boxed{\text{Fixing } a_0 \mapsto b \text{ reduces the dimension by one.}}$

One problem in $d$ dimensions is reduced to $k$ problems in $d-1$ dimensions.

# Three Dimensions [Akutsu 1995]

PRUNE by distance from the origin. If the points lie in a plane or on a line → DIMENSION REDUCTION.

Compute the convex hull.
If there are vertices of different degrees → PRUNE

The number $n$ of vertices is reduced to $\le n/2$. RESTART.

All $n$ vertices have now degree 3, 4, or 5.
There are $f = \frac{n}{2} + 2$ or $f = n + 2$ or $f = \frac{3n}{2} + 2$ faces.

If the face degrees are not all equal
→ switch to the centroids of the faces and PRUNE them.

$n$ is reduced to $\le \frac{3n}{4} + 1$. RESTART.

Now $P(A)$ must have the graph of a Platonic solid. → $n \le 20$.
→ DIMENSION REDUCTION.

# Three Dimensions [Akutsu 1995]

PRUNE by distance from the origin. If the points lie in a plane or on a line → DIMENSION REDUCTION.

Compute the convex hull. ← $\boxed{O(|A|\log|A|) \text{ time}}$

If there are vertices of different degrees → PRUNE

The number $n$ of vertices is reduced to $\le n/2$. RESTART.

All $n$ vertices have now degree 3, 4, or 5.

There are $f = \frac{n}{2} + 2$ or $f = n + 2$ or $f = \frac{3n}{2} + 2$ faces.

If the face degrees are not all equal

→ switch to the centroids of the faces and PRUNE them.

$n$ is reduced to $\le \frac{3n}{4} + 1$. RESTART.

graph-theoretic pruning

TIME =

$O(n\log n) + O(\frac{3}{4}n\log\frac{3}{4}n) + O((\frac{3}{4})^2 n\log((\frac{3}{4})^2 n)) + \ldots$

$= O(n\log n)$

# Three Dimensions [Akutsu 1995]

PRUNE by distance from the origin. If the points lie in a plane or on a line → DIMENSION REDUCTION.

Compute the convex hull.

If there are vertices of different degrees → PRUNE

The number $n$ of vertices is reduced to $\leq n/2$. RESTART.

All $n$ vertices have now degree 3, 4, or 5.

There are $f = \frac{n}{2} + 2$ or $f = n + 2$ or $f = \frac{3n}{2} + 2$ faces.

If the face degrees are not all equal

→ switch to the centroids of the faces and PRUNE them.

$n$ is reduced to $\leq \frac{3n}{4} + 1$. RESTART.

Now $P(A)$ must have the graph of a Platonic solid. $\rightarrow n \leq 20$.

→ DIMENSION REDUCTION.

# Three Dimensions [Akutsu 1995]

PRUNE by distance from the origin. If the points lie in a plane or on a line → DIMENSION REDUCTION.

Canonical point sets in 3d:

We get $\leq 20$ two-dimensional projected point sets.
Rotate the plane to the $x$-$y$-plane.
Compute the canonical 2-d point set.
$\leq 20$ candidates for canonical 3d point sets:
Choose the lex-smallest one.

Now $P(A)$ must have the graph of a Platonic solid. $\rightarrow n \leq 20.$
→ DIMENSION REDUCTION.

# PRUNING/CONDENSING in general

Function $f(A) = A'$, $A' \not\subseteq \{0\}$, equivariant under rotations $R$:

$$f(RA) = RA'$$

$A'$ has *all symmetries* of $A$ (and maybe more).

Primary goal: $|A'| \le |A| \cdot c$, $c < 1$.

If there is a chance, PRUNE and start from scratch with $A'$ instead of $A$.

Ultimate goal: $|A| \le$ const

# Condensing on the 2-Sphere

Use some more geometric pruning to get:

Equivariant condensation on the 2-sphere:

Input: $A \subseteq \mathbb{S}^2$.
Output: $A' \subseteq \mathbb{S}^2$, $|A'| \le \min\{|A|, 12\}$, $A' = f(A)$ equivariant.

5 possibilities:

- $A' =$ vertices of a regular icosahedron
- $A' =$ vertices of a regular octahedron
- $A' =$ vertices of a regular tetrahedron
- $A' =$ two antipodal points, or
- $A' =$ a single point.

# Dimension d

Dimension reduction without pruning:

Pick $a_0 \in A$. Try $a_0 \mapsto b$ for all $b \in B$ ($n$ possibilities).

$\rightarrow O(n^{d-2} \log n)$ time [ Alt, Mehlhorn, Wagener, Welzl 1988 ]

# Dimension d

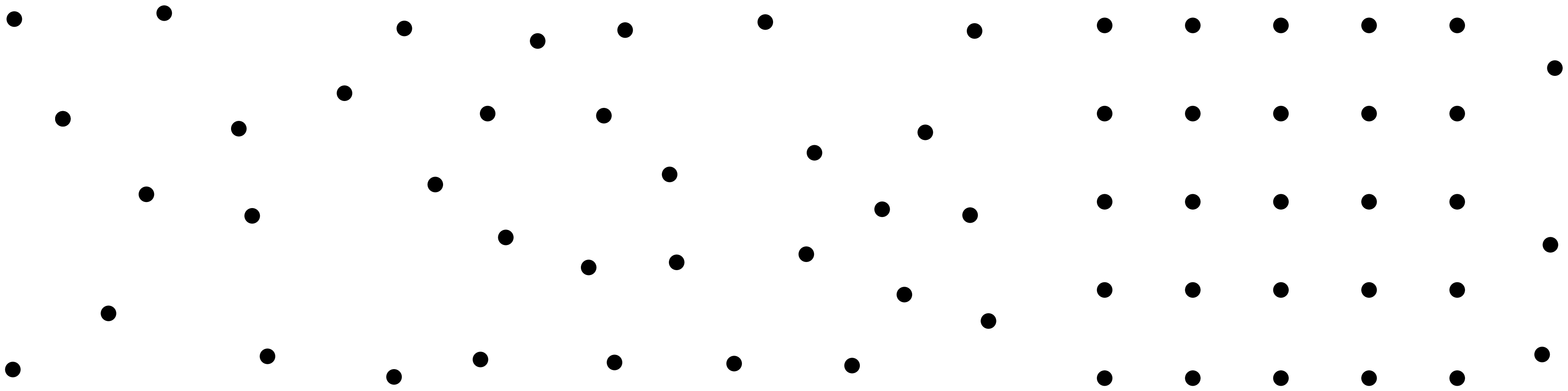
Dimension reduction without pruning:

Pick $a_0 \in A$. Try $a_0 \mapsto b$ for all $b \in B$ ($n$ possibilities).

$\rightarrow O(n^{d-2} \log n)$ time [ Alt, Mehlhorn, Wagener, Welzl 1988 ]

Closest pairs $(a, a')$: [Matoušek ≈ 1998]

minimum distance $\delta := \|a - a'\|$ among all pairs of vertices



# Dimension d

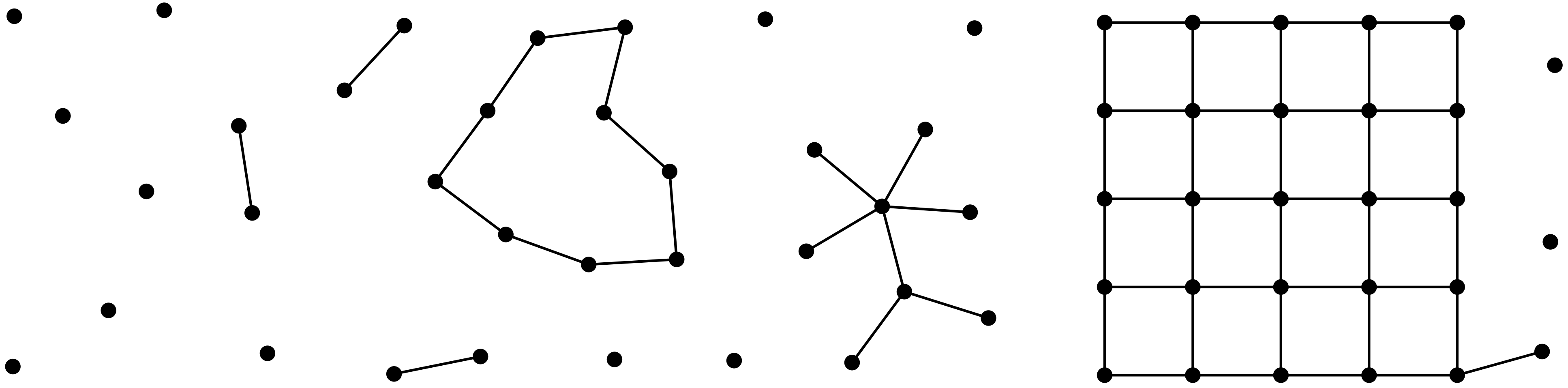
Dimension reduction without pruning:

Pick $a_0 \in A$. Try $a_0 \mapsto b$ for all $b \in B$ ($n$ possibilities).

$\rightarrow O(n^{d-2} \log n)$ time [ Alt, Mehlhorn, Wagener, Welzl 1988 ]

Closest pairs $(a, a')$: [Matoušek ≈ 1998]

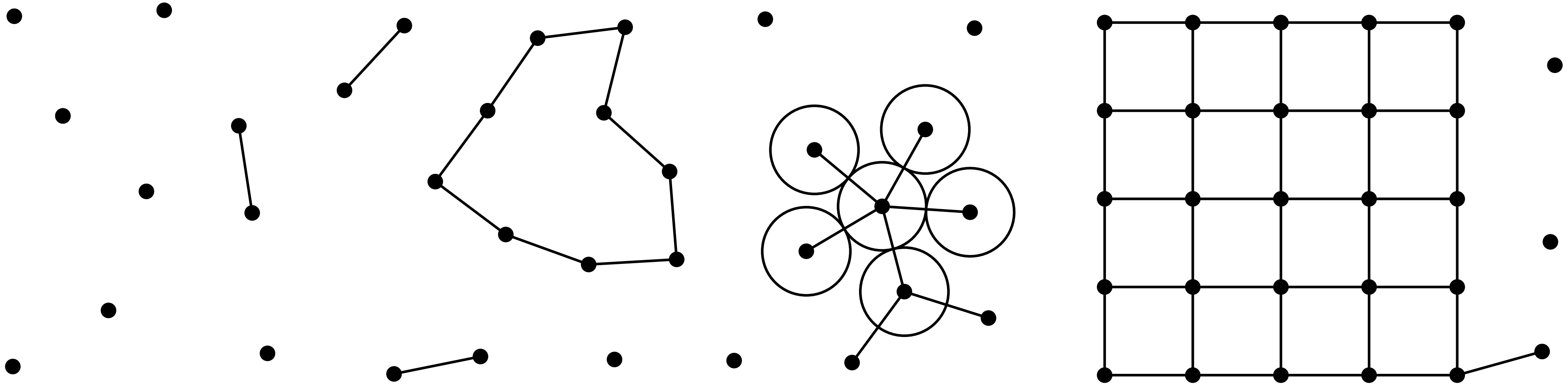
minimum distance $\delta := \|a - a'\|$ among all pairs of vertices



# Dimension d

Dimension reduction without pruning:
Pick $a_0 \in A$. Try $a_0 \mapsto b$ for all $b \in B$ ($n$ possibilities).
$\rightarrow O(n^{d-2} \log n)$ time [ Alt, Mehlhorn, Wagener, Welzl 1988 ]

Closest pairs $(a, a')$: [Matoušek ≈ 1998]
minimum distance $\delta := \|a - a'\|$ among all pairs of vertices



Degree $\leq$ the *kissing number* $K_d$ (packing argument).

All closest pairs can be computed in $O(n \log n)$ time ($d$ fixed).
[ Bentley and Shamos, STOC 1976 ]

# Dimension d

Dimension reduction without pruning:

Pick $a_0 \in A$. Try $a_0 \mapsto b$ for all $b \in B$ ($n$ possibilities).

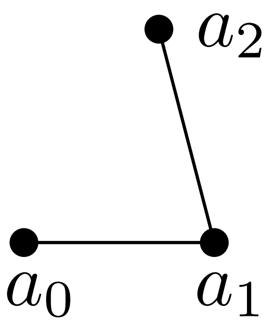
$\rightarrow O(n^{d-2} \log n)$ time [ Alt, Mehlhorn, Wagener, Welzl 1988 ]

Pick a closest pair $a_0 a_1$ in $A$. Try $(a_0, a_1) \mapsto (b, b')$ for all closest pairs $(b, b')$ in $B$.

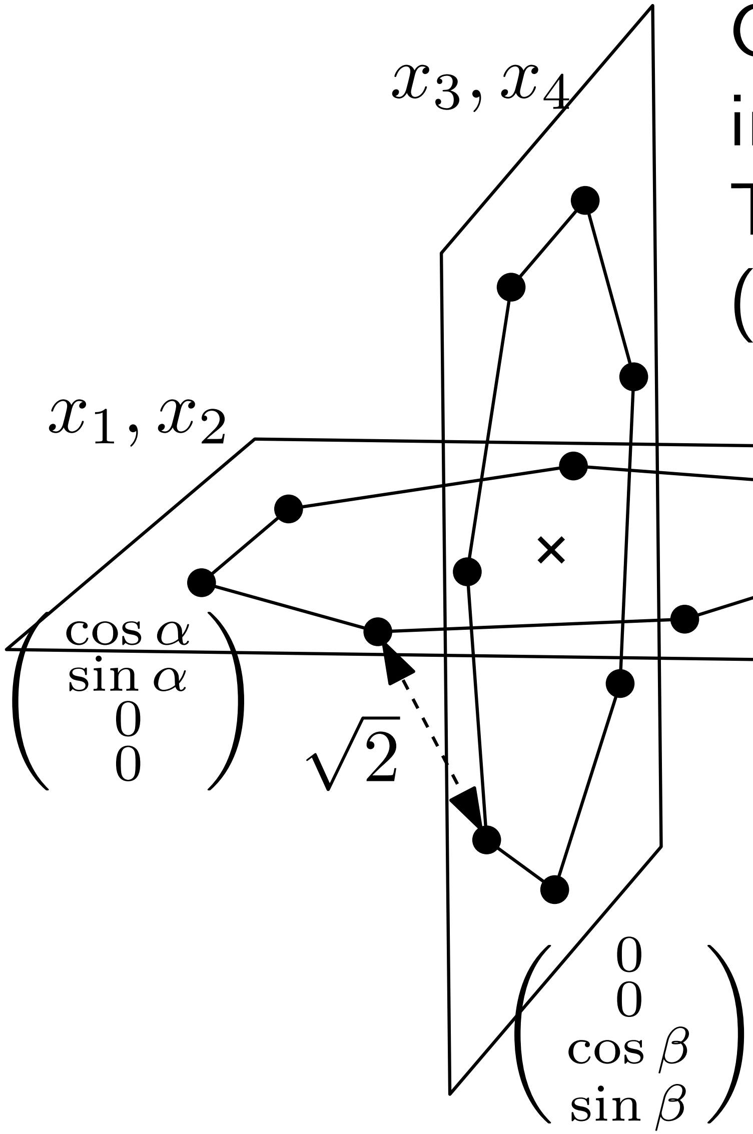
$O(n)$ possibilities, reducing the dimension by two.

$\rightarrow O(n^{\lfloor d/2 \rfloor} \log n)$ time [ Matoušek ≈ 1998 ]

Further improvement: Find a "closest triplet" . . .



# Life in Four Dimensions


$x_3, x_4$
$x_1, x_2$
×
$\begin{pmatrix} \cos\alpha \\ \sin\alpha \\ 0 \\ 0 \end{pmatrix}$
$\sqrt{2}$
$\begin{pmatrix} 0 \\ 0 \\ \cos\beta \\ \sin\beta \end{pmatrix}$

Consider two regular $n$-gons
in the $x_1x_2$-plane and the $x_3x_4$-plane.
There are $\Theta(n^2)$ "closest triplets".
(Triplets on the same $n$-gon are not useful.)

The convex hull has
$\Theta(n^2)$ edges and facets.

[ Brass and Knauer 2002 ]
Point sets in orthogonal subspaces
are the only problematic case;
they can be treated specially.

→ $O(n^{\lceil d/3 \rceil} \log n)$ time

# Use Randomness in $d$ Dimensions

- Random sampling
- Birthday paradox
- Closest pairs

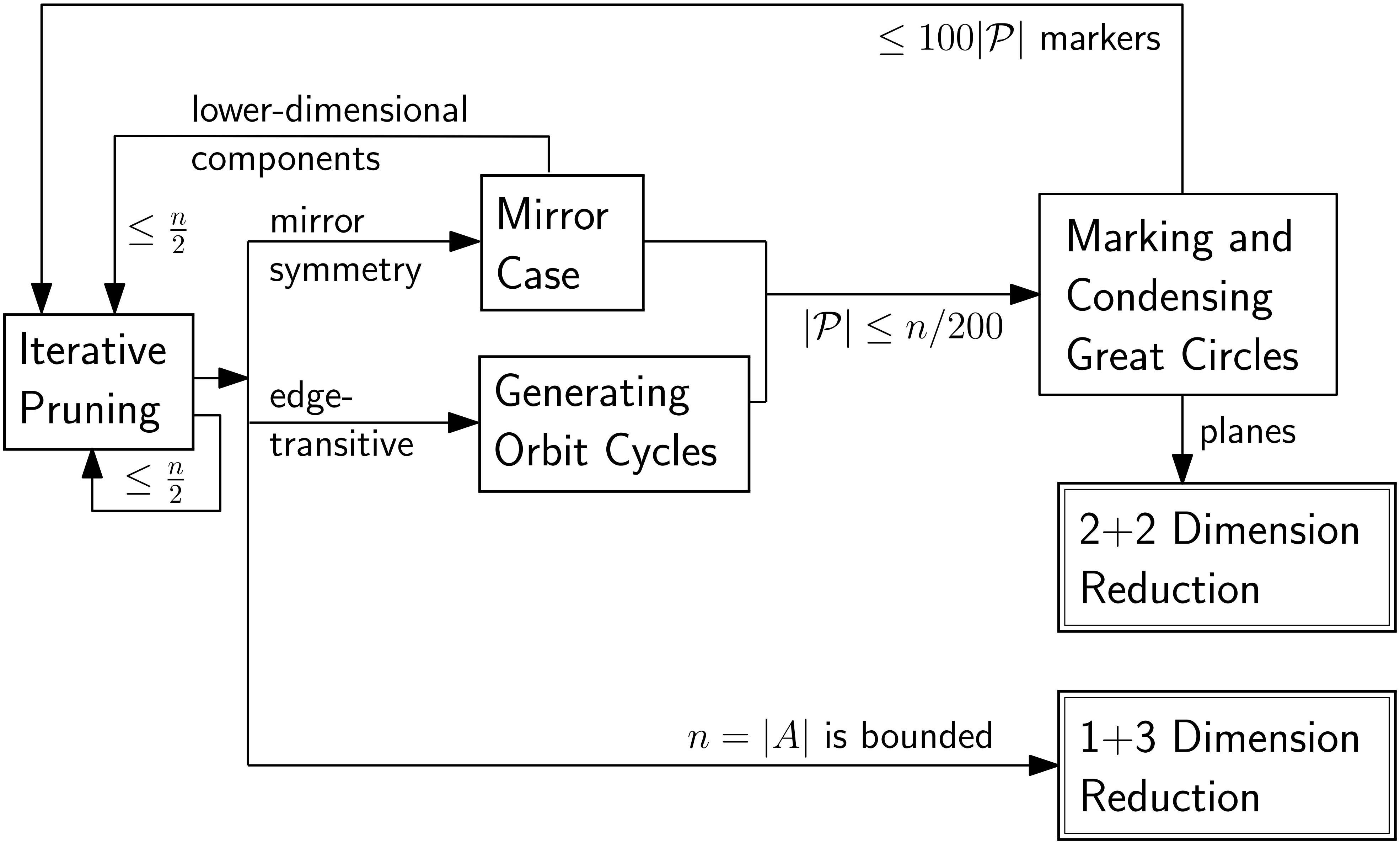
→ Monte Carlo algorithm,

$$O(n^{\lfloor d/2 \rfloor/2} \log n) \text{ time}, \quad O(n^{\lfloor d/2 \rfloor/2}) \text{ space}$$

[Akutsu 1998 + improvement by J. Matoušek, personal communication]

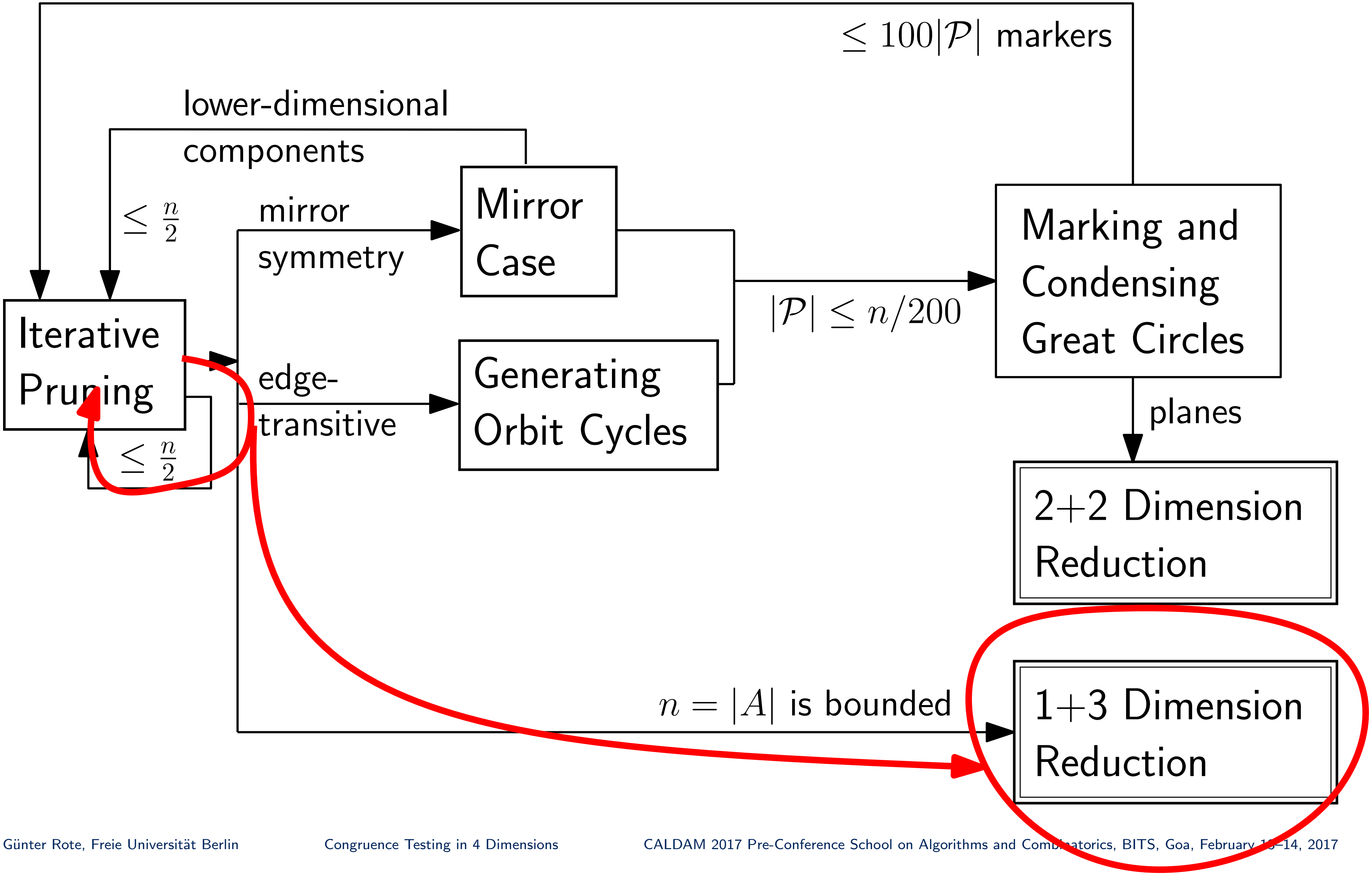
# 4 Dimensions: Algorithm Overview

joint work with Heuna Kim



# 4 Dimensions: Algorithm Overview

joint work with Heuna Kim



# Initialization: Closest-Pair Graph

1) PRUNE by distance from the origin.

- $\Longrightarrow$ we can assume that $A$ lies on the 3-sphere $\mathbb{S}^3$.

2) Compute the closest pair graph

$$G(A) = (A, \{\, uv : \|u - v\| = \delta \,\})$$

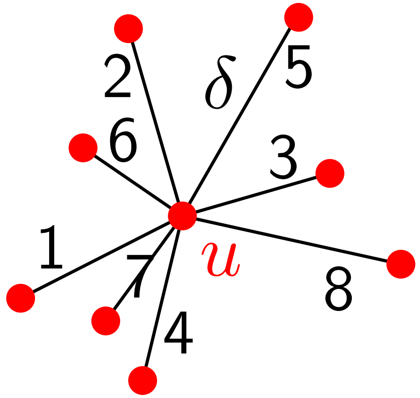
where $\delta :=$ the distance of the closest pair, in $O(n \log n)$ time.

- We can assume that $\delta$ is SMALL: $\delta \le \delta_0 := 0.0005$. (Otherwise, $|A| \le n_0$, by a packing argument.)

# Everything Looks the Same!

By the PRUNING principle, we can assume that all points look locally the same:

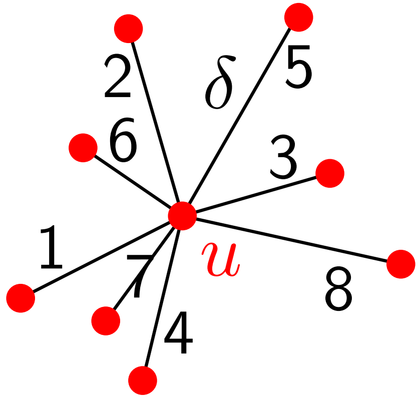
- All points have congruent neighborhoods in $G(A)$.
  (The neighbors of $u$ lie on a 2-sphere in $\mathbb{S}^3$;
  There are at most $K_3 = 12$ neighbors.)



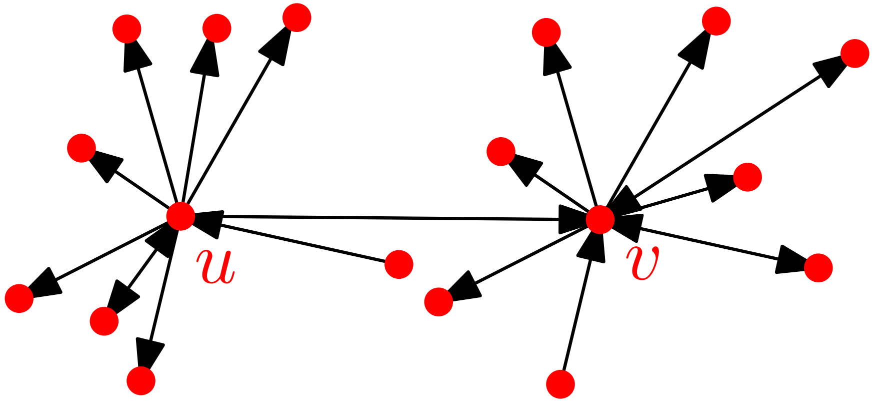
# Everything Looks the Same!

By the PRUNING principle, we can assume that all points look locally the same:

- All points have congruent neighborhoods in $G(A)$.
  (The neighbors of $u$ lie on a 2-sphere in $\mathbb{S}^3$;
  There are at most $K_3 = 12$ neighbors.)

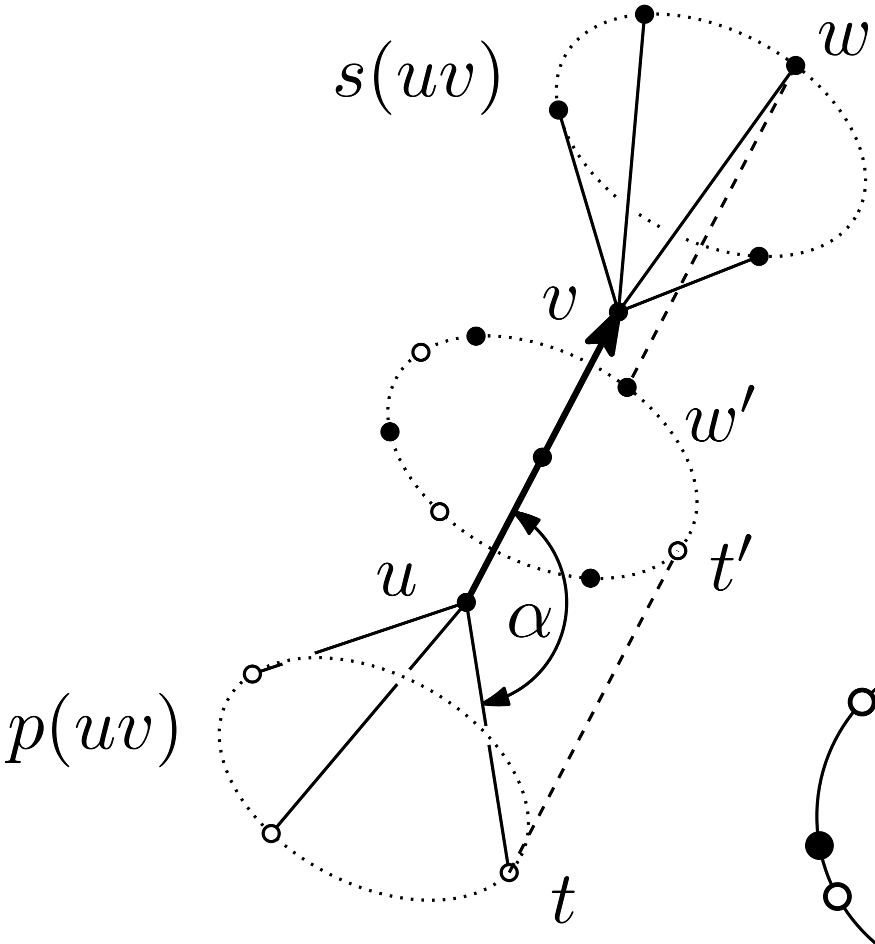


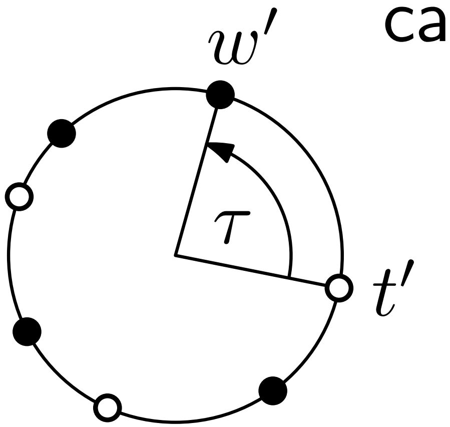
- Make a directed graph $D$ from $G(A)$
  and PRUNE its arcs $uv$ by
  the joint neighborhood of $u$ and $v$.



# Further Pruning

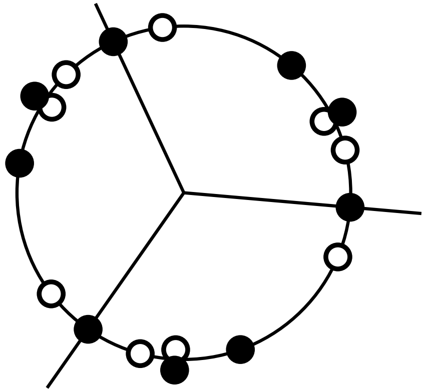
Pick some $\alpha$. $\quad s(uv) := \{vw : vw \in E, \angle uvw = \alpha\}$



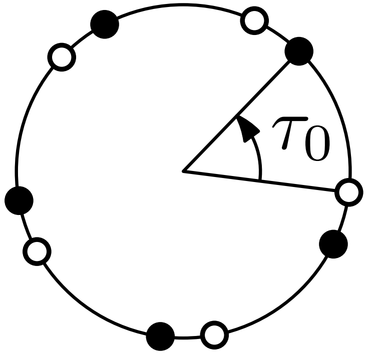


torsion angle $\tau$

canonical directions



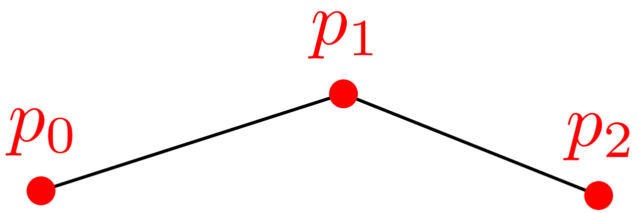
can PRUNE arcs from $s(u,v)$



For every path $tuv$ with $\angle tuv = \alpha$, $\exists\ vw$ with $\angle uvw = \alpha$ and torsion $\tau_0$.

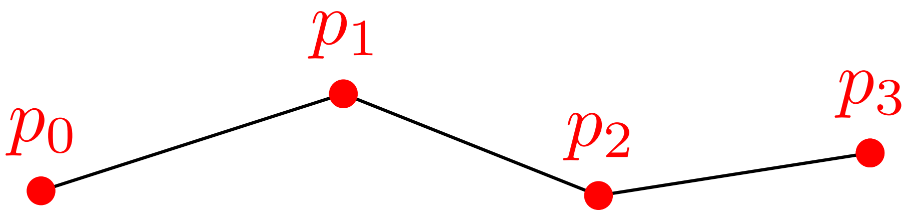
# Construct Orbit Cycles

! For every path $p_i p_{i+1} p_{i+2}$ with $\angle p_i p_{i+1} p_{i+2} = \alpha$, $\exists\ p_{i+3}$ with $\angle p_{i+1} p_{i+2} p_{i+3} = \alpha$ and torsion $\tau_0$.



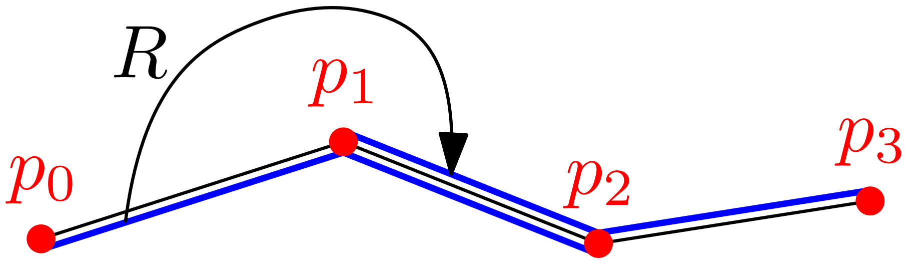
# Construct Orbit Cycles

! For every path $p_i p_{i+1} p_{i+2}$ with $\angle p_i p_{i+1} p_{i+2} = \alpha$, $\exists\ p_{i+3}$ with $\angle p_{i+1} p_{i+2} p_{i+3} = \alpha$ and torsion $\tau_0$.



# Construct Orbit Cycles

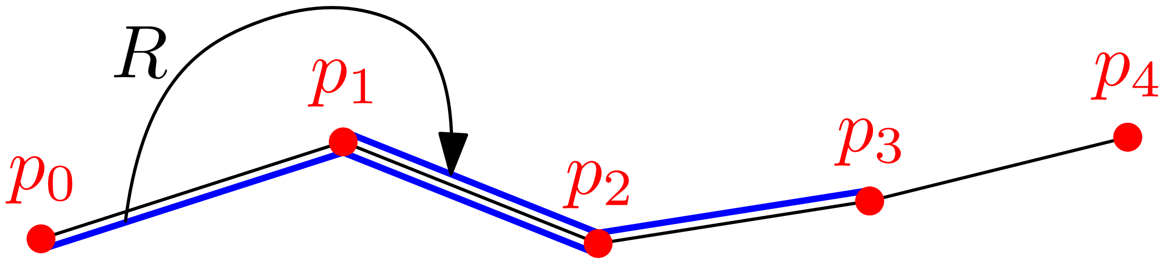
! For every path $p_i p_{i+1} p_{i+2}$ with $\angle p_i p_{i+1} p_{i+2} = \alpha$,
$\exists\ p_{i+3}$ with $\angle p_{i+1} p_{i+2} p_{i+3} = \alpha$ and torsion $\tau_0$.



$$R(p_0, p_1, p_2) = (p_1, p_2, p_3)$$

# Construct Orbit Cycles

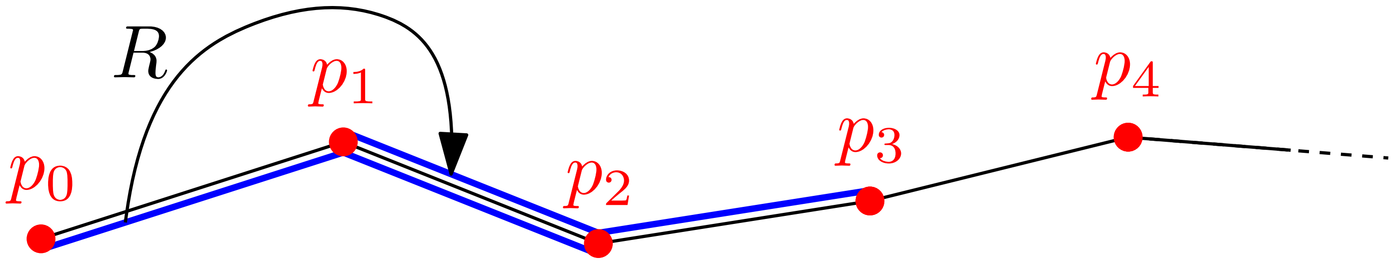
! For every path $p_i p_{i+1} p_{i+2}$ with $\angle p_i p_{i+1} p_{i+2} = \alpha$,
$\exists\ p_{i+3}$ with $\angle p_{i+1} p_{i+2} p_{i+3} = \alpha$ and torsion $\tau_0$.



$$R(p_0, p_1, p_2) = (p_1, p_2, p_3)$$
$$R(p_0, p_1, p_2, p_3) = (p_1, p_2, p_3, p_4)$$

# Construct Orbit Cycles

! For every path $p_i p_{i+1} p_{i+2}$ with $\angle p_i p_{i+1} p_{i+2} = \alpha$,
$\exists\ p_{i+3}$ with $\angle p_{i+1} p_{i+2} p_{i+3} = \alpha$ and torsion $\tau_0$.



$$R(p_0, p_1, p_2) = (p_1, p_2, p_3)$$
$$R(p_0, p_1, p_2, p_3) = (p_1, p_2, p_3, p_4)$$
$$R(p_1, p_2, p_3, p_4) = (p_2, p_3, p_4, p_5)$$
$$\ldots$$

$Rp_i = p_{i+1}$: The orbit of $p_0$ under $R$, a helix

# Rotations in 4 Dimensions

Freie Universität Berlin

$$R = \begin{pmatrix} \cos\varphi & -\sin\varphi & 0 & 0 \\ \sin\varphi & \cos\varphi & 0 & 0 \\ 0 & 0 & \cos\psi & -\sin\psi \\ 0 & 0 & \sin\psi & \cos\psi \end{pmatrix} = \begin{pmatrix} R_\varphi & 0 \\ 0 & R_\psi \end{pmatrix}$$

in some appropriate coordinate system.

$\varphi \neq \pm\psi$: $\to$ unique decomposition $\mathbb{R}^4 = P \oplus Q$ into two completely orthogonal 2-dimensional *axis planes* $P$ and $Q$

$\varphi = \pm\psi$: isoclinic rotations

The orbit of a point $a_0 = (x_1, x_2, x_3, x_4)$ lies on a *helix* on a *flat torus* $C_r \times C_s$, with $r = \sqrt{x_1^2 + x_2^2}$, $s = \sqrt{x_3^2 + x_4^2}$

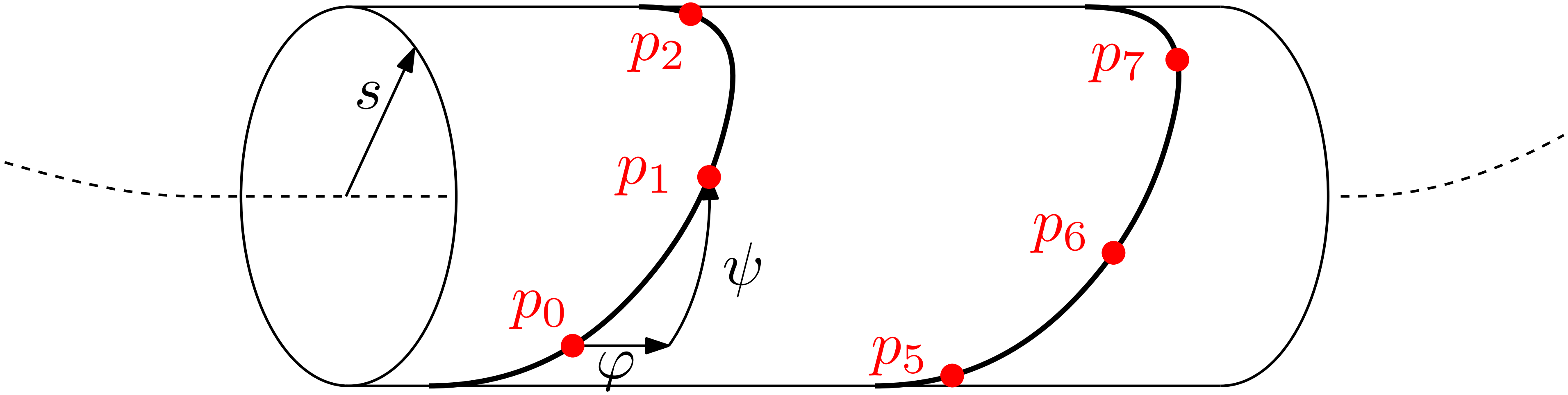
$\uparrow$

circle with radius $r$

# Rotations in 4 Dimensions

$$R = \begin{pmatrix} \cos\varphi & -\sin\varphi & 0 & 0 \\ \sin\varphi & \cos\varphi & 0 & 0 \\ 0 & 0 & \cos\psi & -\sin\psi \\ 0 & 0 & \sin\psi & \cos\psi \end{pmatrix} = \begin{pmatrix} R_\varphi & 0 \\ 0 & R_\psi \end{pmatrix}$$

The orbit of a point $p_0 = (x_1, y_1, x_2, y_2)$ lies on a *helix* on a *flat torus* $C_r \times C_s$, with $r = \sqrt{x_1^2 + y_1^2}$, $s = \sqrt{x_2^2 + y_2^2}$

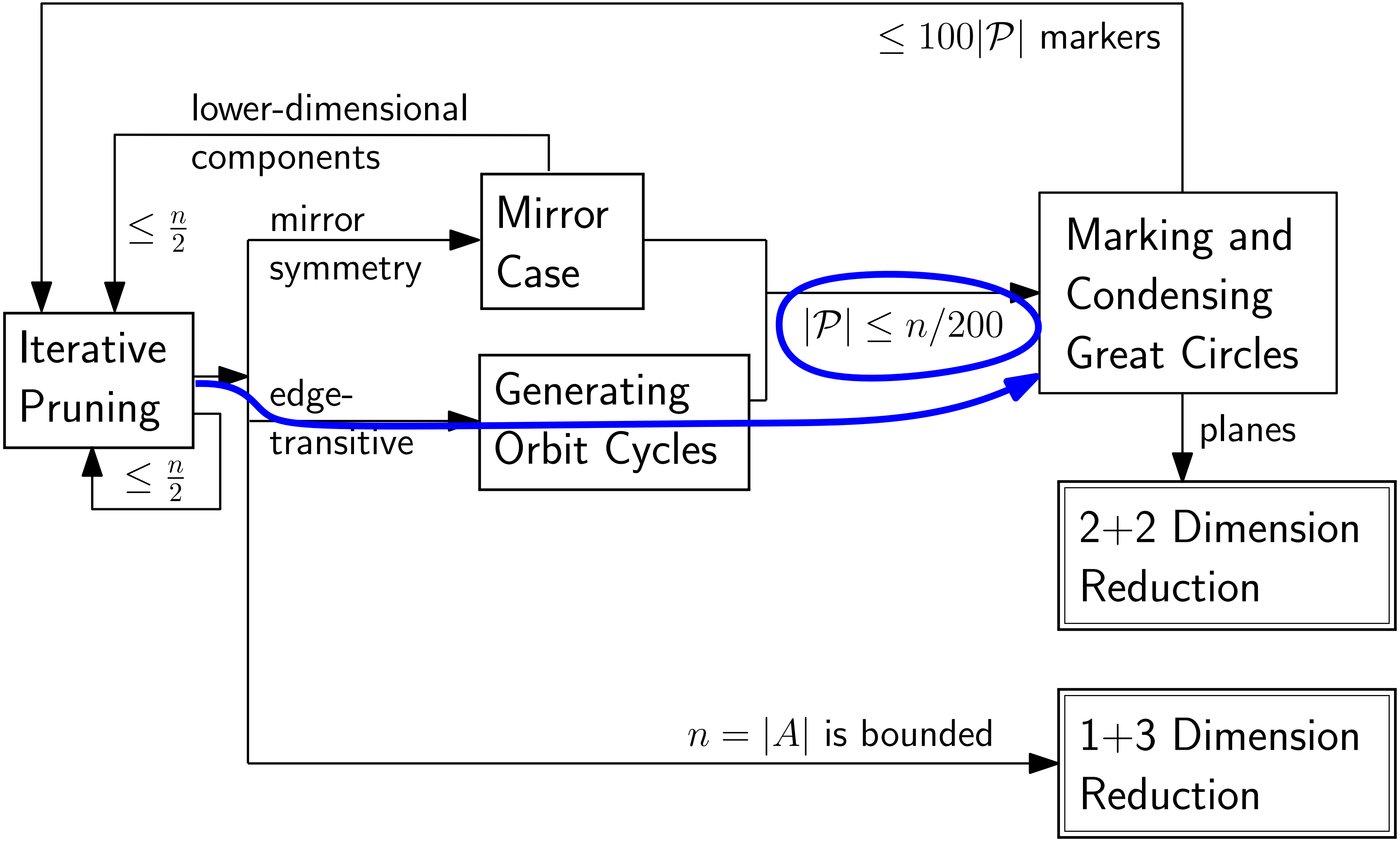


# Planes in 4 Dimensions

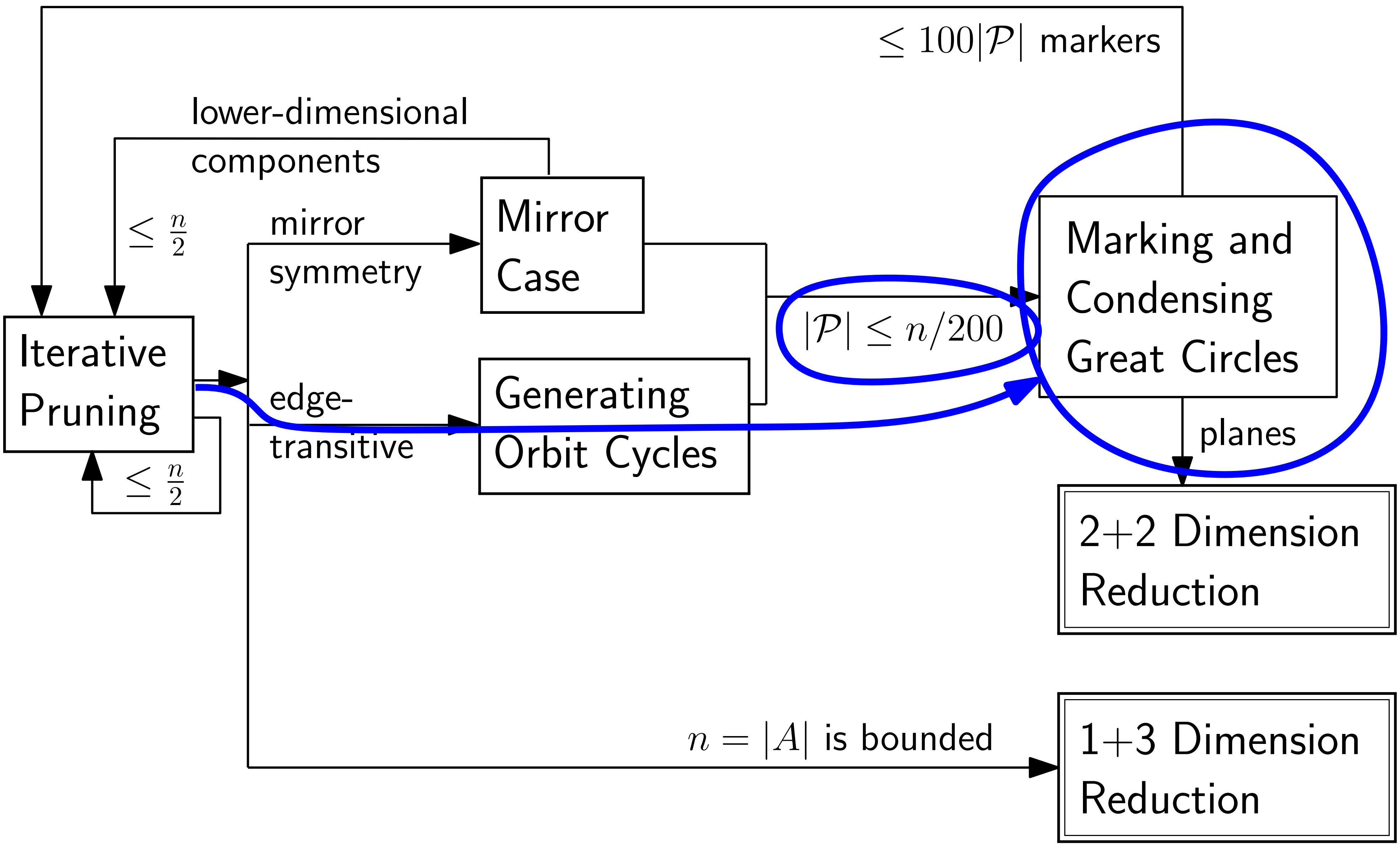
- Every point lies on $\leq 60$ orbit cycles.
- Every orbit cycle contains $\geq 12000$ points, because $\delta$ is small.
- Every orbit cycle generates 1 plane (corresponding to the smaller of $\varphi$ and $\psi$.)

$\Longrightarrow$ a collection of $\leq n/200$ planes (or: great circles)

# Algorithm Overview



# Algorithm Overview



Iterative Pruning

Mirror Case

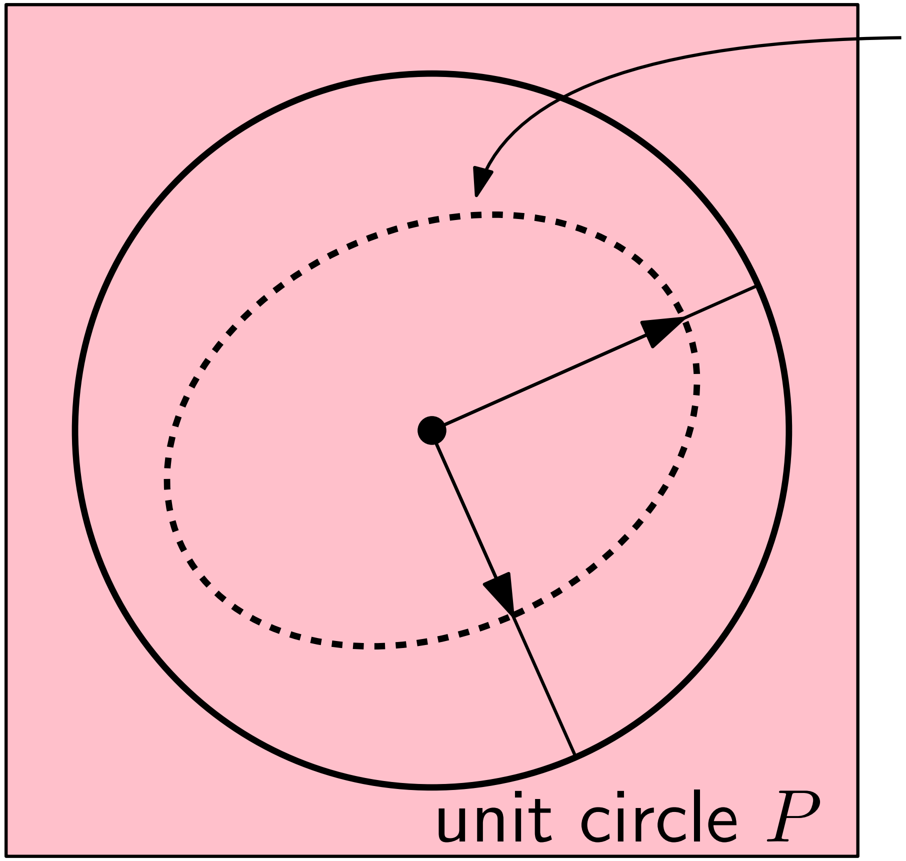
Generating Orbit Cycles

Marking and Condensing Great Circles

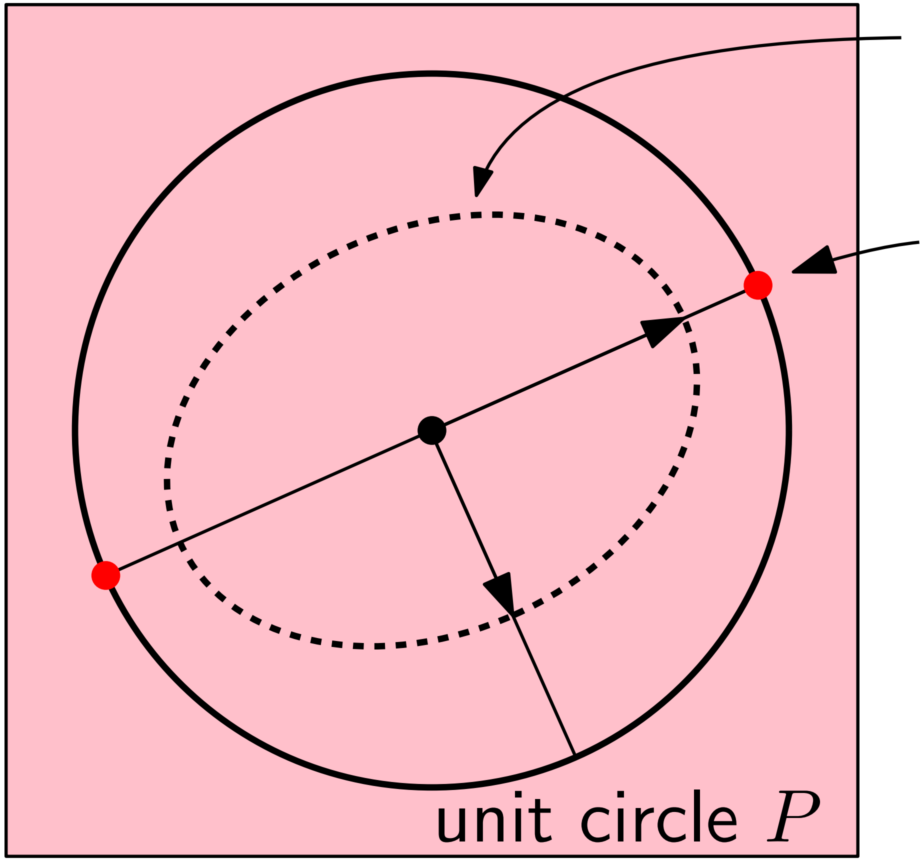
2+2 Dimension Reduction

1+3 Dimension Reduction

# Marking Points on Great Circles



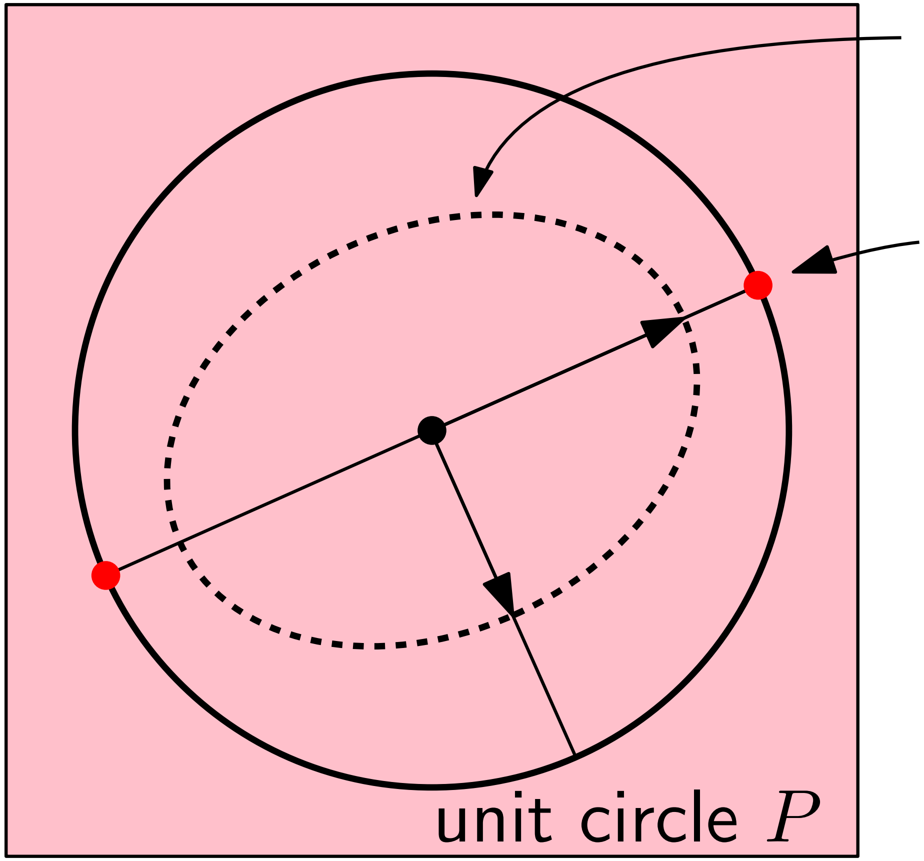
# Marking Points on Great Circles



projection of another unit circle $Q$

IDEA: mark those two points in $P$

# Marking Points on Great Circles



projection of ~~another unit circle~~ $Q$
a neighbor of $P$

IDEA: mark those two points in $P$

IDEA 2: Construct the closest-pair graph in the space of great circles, in $O(n \log n)$ time.
Every plane has at most $K_5 \leq 44$ neighbors.

# Plücker coordinates

planes in 4-space $\Leftrightarrow$ great circles on $\mathbb{S}^3$ $\Leftrightarrow$ a.k.a. lines in $\mathbb{R}P^3$

plane through $(x_1, y_1, x_2, y_2)$ and $(x_1', y_1', x_2', y_2')$ :

$$(v_1, \dots, v_6) = \left( \begin{vmatrix} x_1 & y_1 \\ x_1' & y_1' \end{vmatrix}, \begin{vmatrix} x_1 & x_2 \\ x_1' & x_2' \end{vmatrix}, \begin{vmatrix} x_1 & y_2 \\ x_1' & y_2' \end{vmatrix}, \begin{vmatrix} y_1 & x_2 \\ y_1' & x_2' \end{vmatrix}, \begin{vmatrix} y_1 & y_2 \\ y_1' & y_2' \end{vmatrix}, \begin{vmatrix} x_2 & y_2 \\ x_2' & y_2' \end{vmatrix} \right)$$

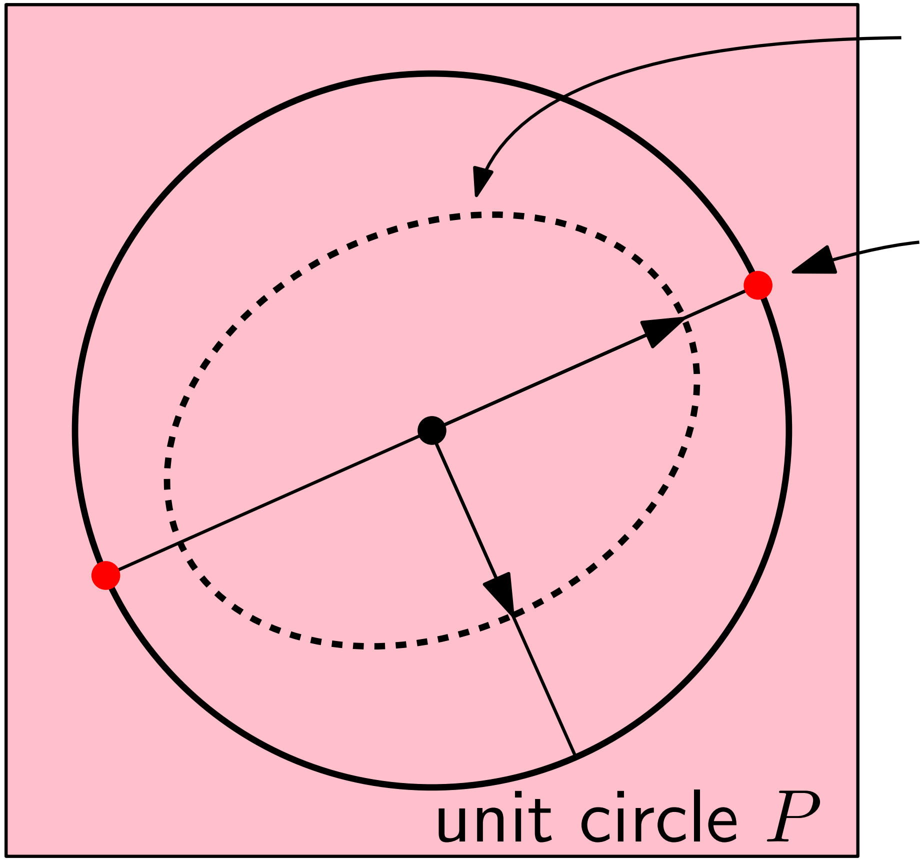
$(v_1, \dots, v_6) \in \mathbb{R}P^5$. [Plücker relations $v_1 v_6 - v_2 v_5 + v_3 v_4 = 0$]

Normalize:
$\rightarrow$ A great circle is represented by two antipodal points on $\mathbb{S}^5$.

This representation is geometrically meaningful:
Distances on $\mathbb{S}^5$ are preserved under rotations of $\mathbb{R}^4$ / $\mathbb{S}^3$.

(Packings of 2-planes in 4-space were considered by [Conway, Hardin and Sloan 1996], with different distances.)

# Marking Points on Great Circles

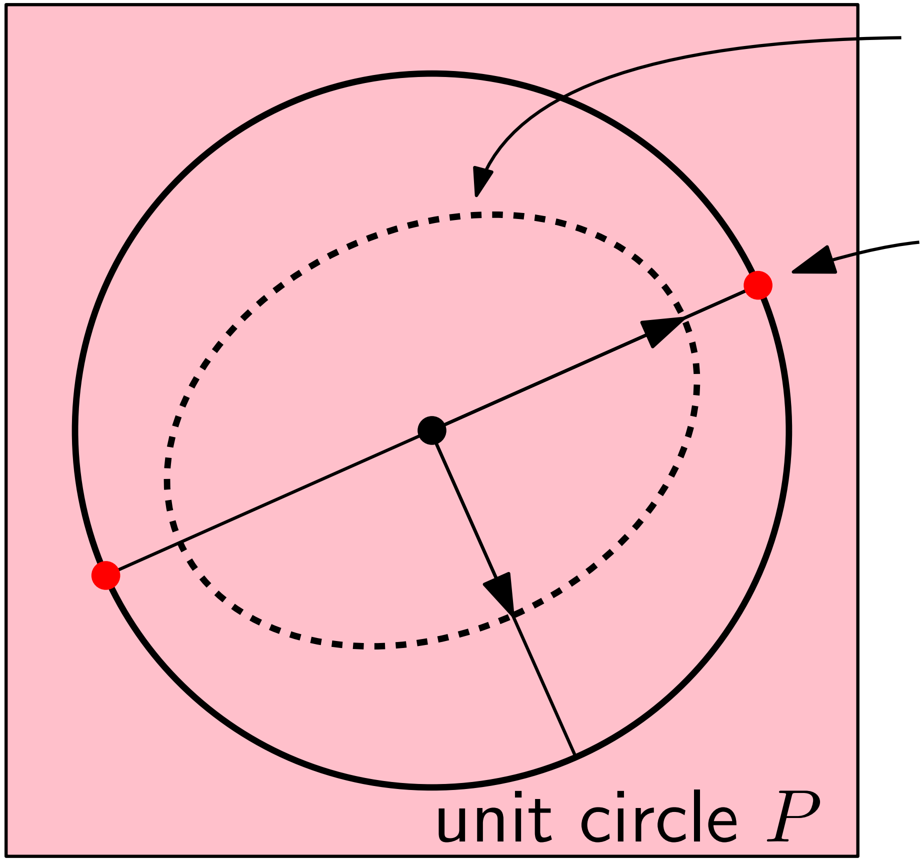


projection of ~~another unit circle~~ $Q$
a neighbor of $P$

IDEA: mark those two points in $P$

IDEA 2: Construct the closest-pair graph in the space of great circles, in $O(n \log n)$ time.
Every plane has at most $K_5 \leq 44$ neighbors.

# Marking Points on Great Circles



projection of ~~another unit circle~~ $Q$
a neighbor of $P$

IDEA: mark those two points in $P$

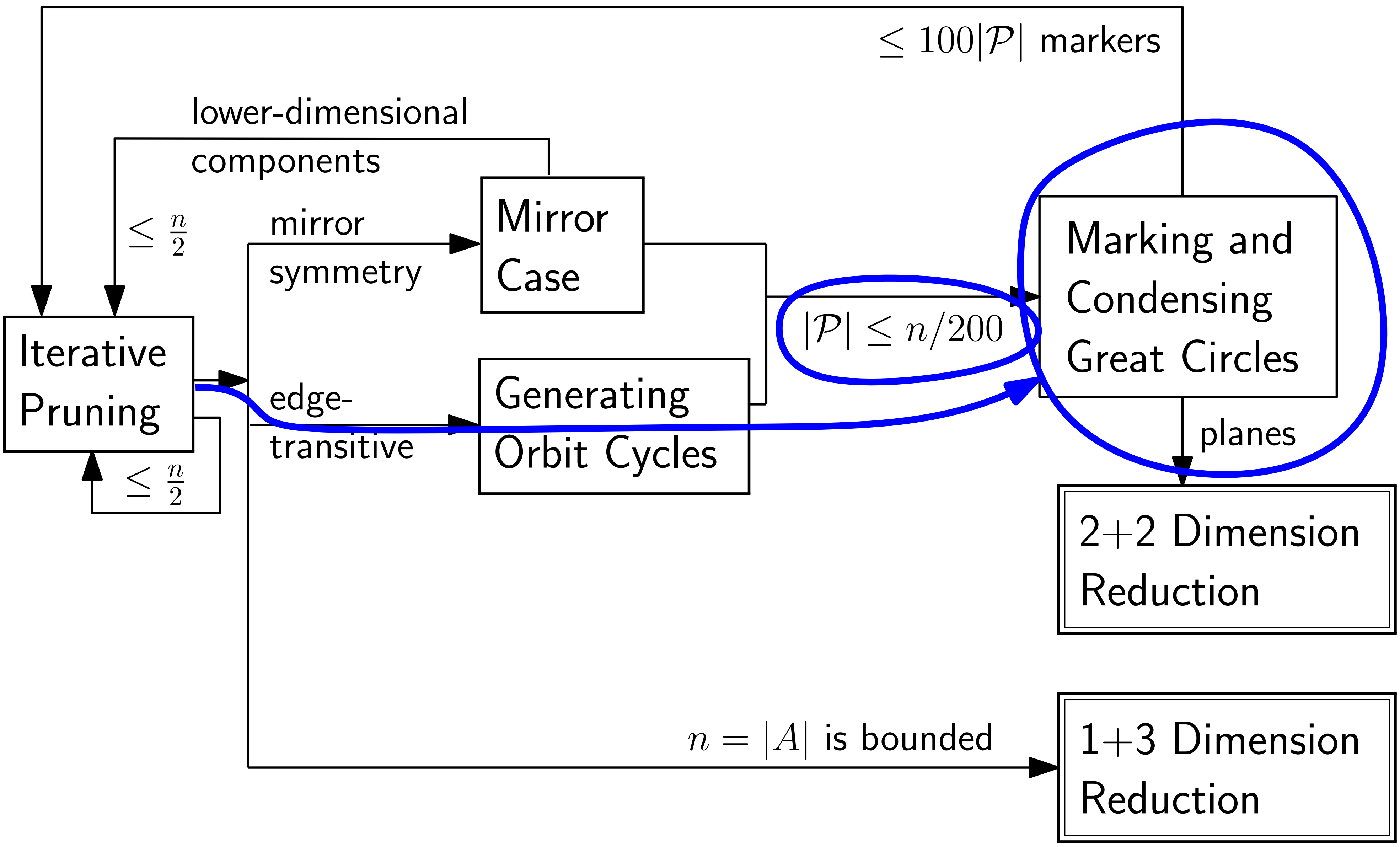
IDEA 2: Construct the closest-pair graph in the space of great circles, in $O(n \log n)$ time.
Every plane has at most $K_5 \leq 44$ neighbors.

$m \leq \frac{n}{200}$ great circles in $\mathbb{R}^4$ $\longrightarrow$ $m$ point pairs on $\mathbb{S}^5$

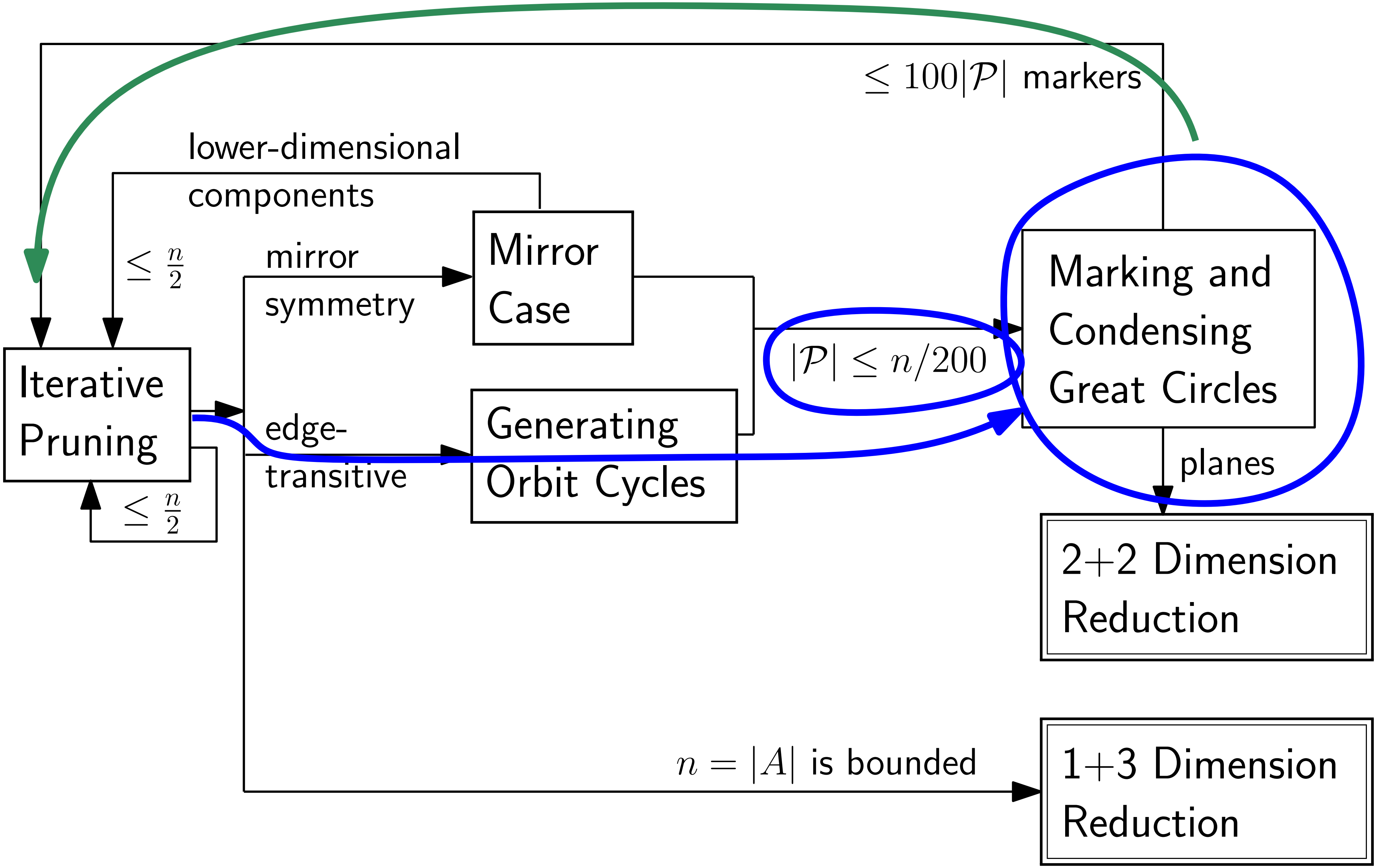
At most 88 points are marked on every great circle.

These points replace $A$. → successful PRUNING

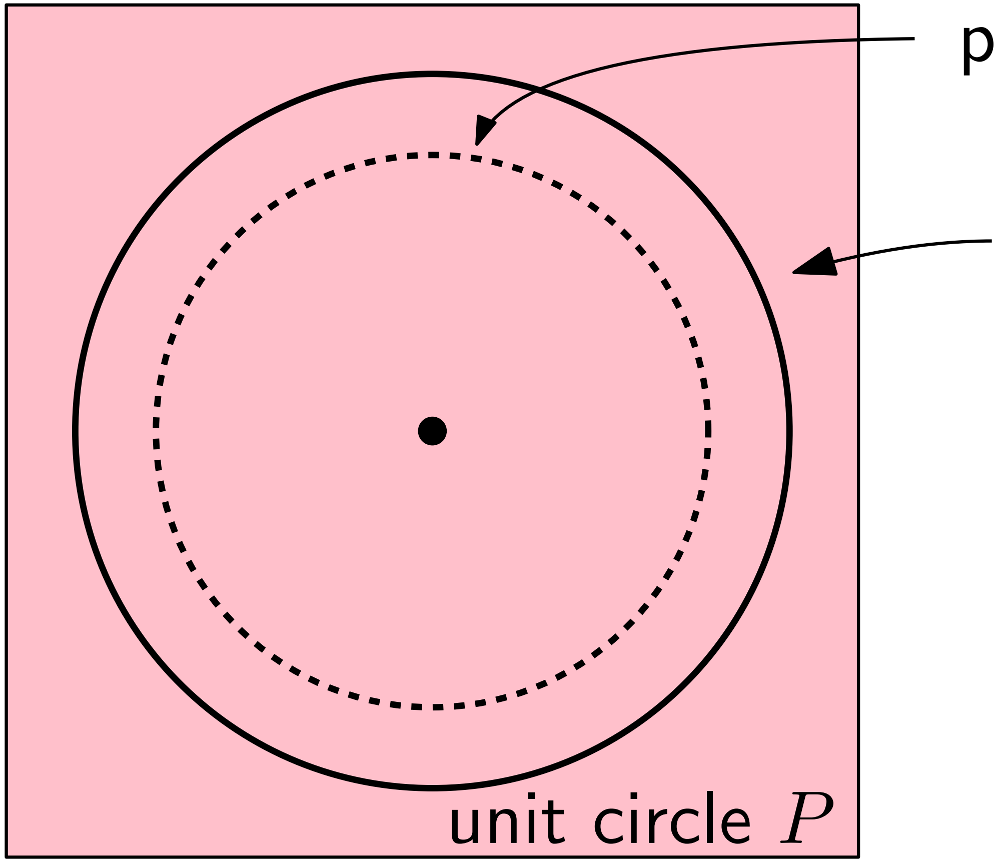
# Algorithm Overview



# Algorithm Overview



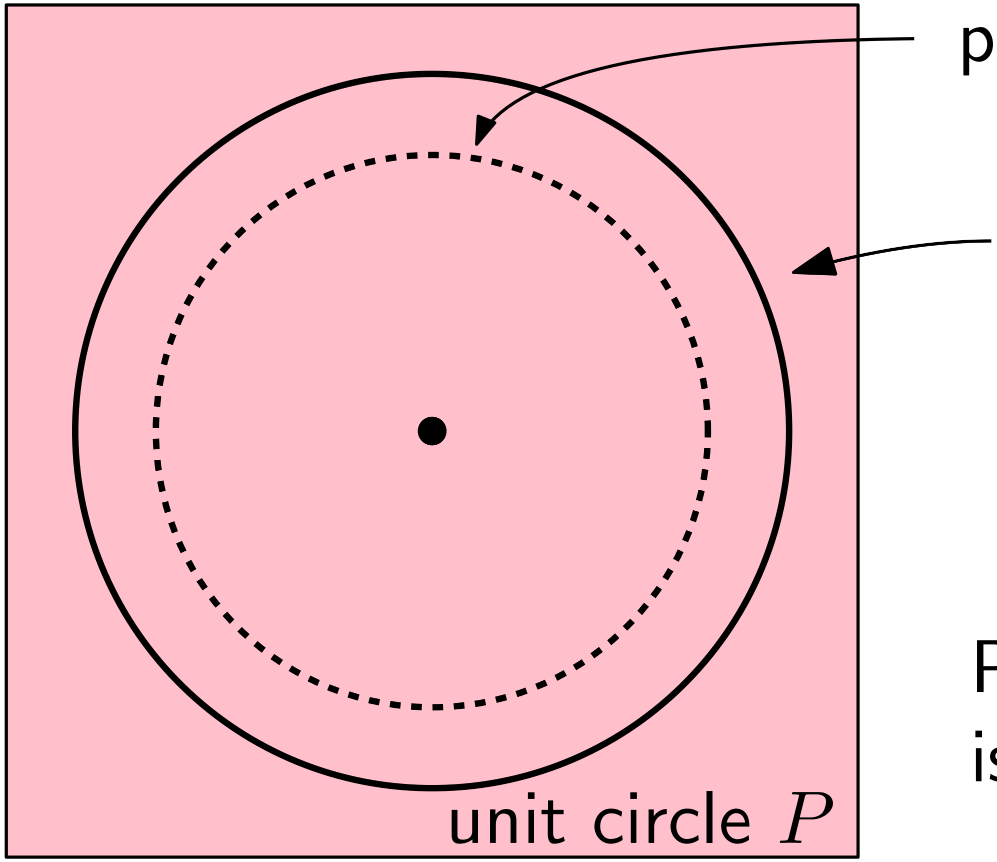
# Isoclinic Planes



projection of a neighbor $Q$ of $P$

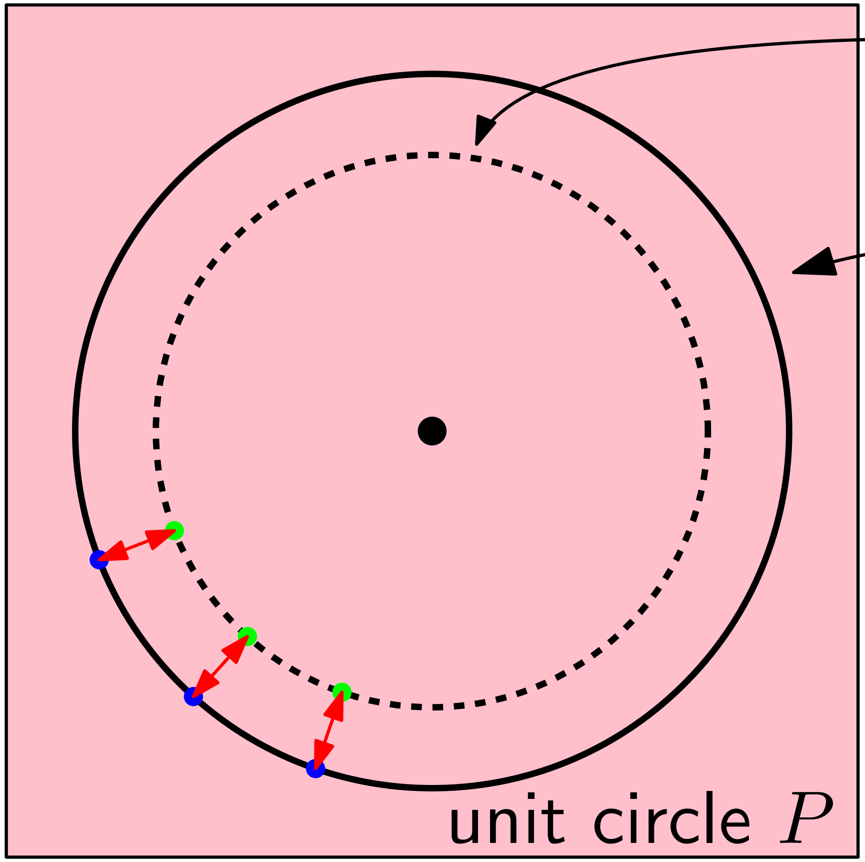
Where to mark??

# Isoclinic Planes



Problem if *all* closest pairs are isoclinic.

# Isoclinic Planes



projection of a neighbor $Q$ of $P$

Where to mark??

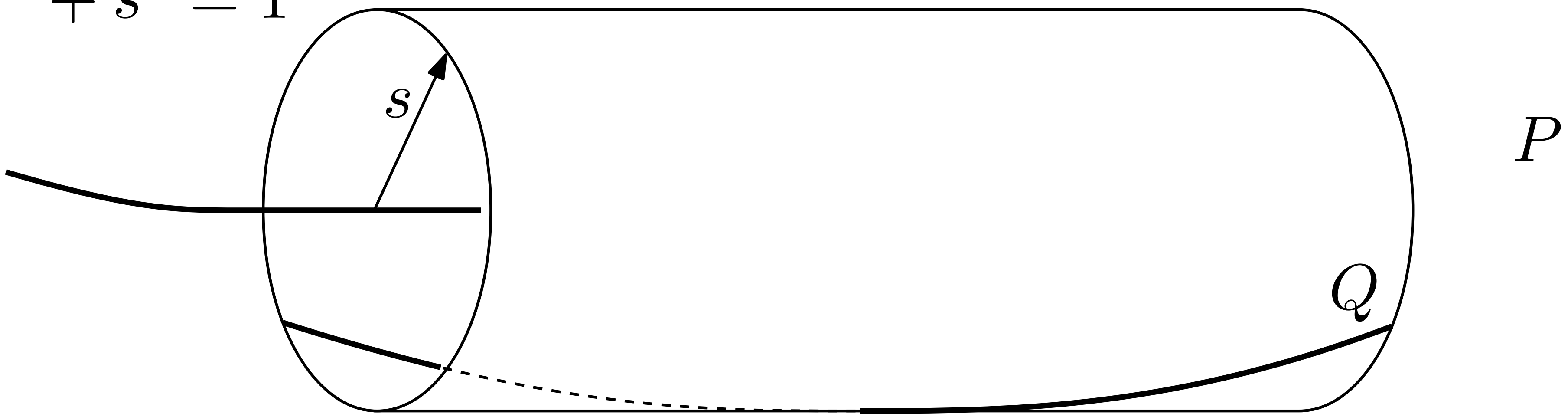
Problem if *all* closest pairs are isoclinic.

Constant distances from one circle to the other.
“Clifford-parallel” $\equiv$ isoclinic

# Clifford-parallel circles

$$P\colon \begin{pmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} \cos t \\ \sin t \\ 0 \\ 0 \end{pmatrix}, \quad Q\colon \begin{pmatrix} r\cos t \\ r\sin t \\ s\cos(\alpha+t) \\ s\sin(\alpha+t) \end{pmatrix}$$

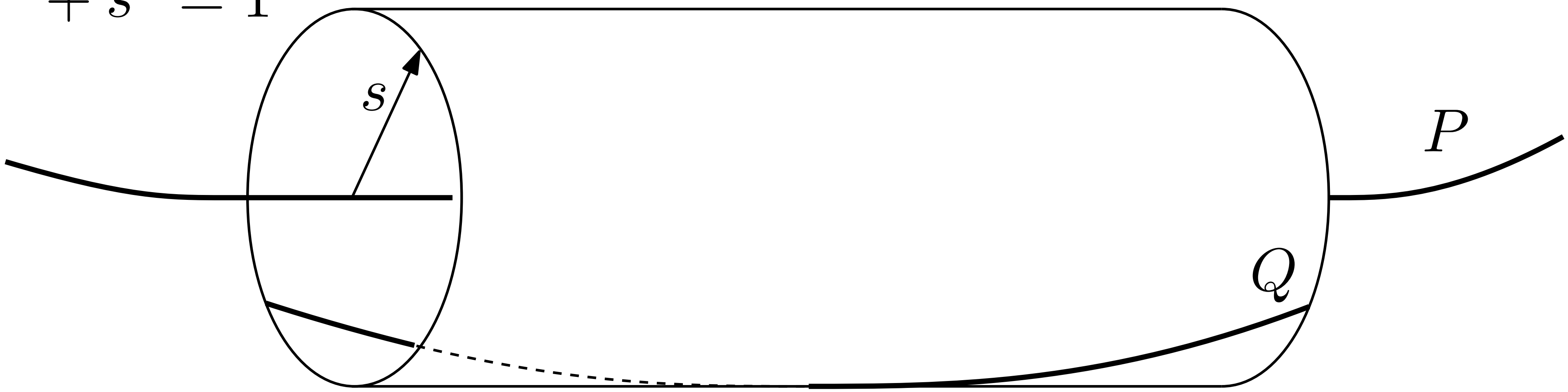
$r^2 + s^2 = 1$



# Clifford-parallel circles

$$P\colon \begin{pmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} \cos t \\ \sin t \\ 0 \\ 0 \end{pmatrix}, \quad Q\colon \begin{pmatrix} r\cos t \\ r\sin t \\ s\cos(\alpha+t) \\ s\sin(\alpha+t) \end{pmatrix}$$

$r^2 + s^2 = 1$



$h(x_1, y_1, x_2, y_2) =$ the *right* Hopf map $h\colon \mathbb{S}^3 \to \mathbb{S}^2$

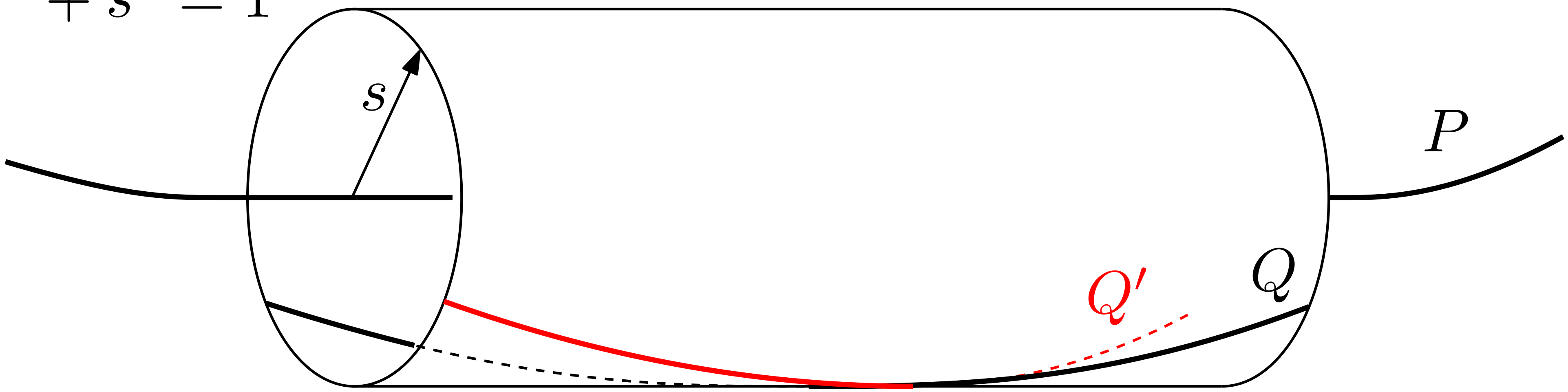
$$\Big(2(x_1y_2 - y_1x_2),\ 2(x_1x_2 + y_1y_2),\ 1 - 2(x_2^2 + y_2^2)\Big)$$

[ Hopf 1931 ]

# Clifford-parallel circles

$$P\colon \begin{pmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} \cos t \\ \sin t \\ 0 \\ 0 \end{pmatrix}, \quad Q\colon \begin{pmatrix} r\cos t \\ r\sin t \\ s\cos(\alpha + t) \\ s\sin(\alpha + t) \end{pmatrix} \qquad Q'\colon \begin{pmatrix} r\cos t \\ r\sin t \\ s\cos(\alpha - t) \\ s\sin(\alpha - t) \end{pmatrix}$$

$r^2 + s^2 = 1$



$h(x_1, y_1, x_2, y_2) =$ the *right* Hopf map $h\colon \mathbb{S}^3 \to \mathbb{S}^2$

$$\Big(2(x_1y_2 - y_1x_2),\ 2(x_1x_2 + y_1y_2),\ 1 - 2(x_2^2 + y_2^2)\Big)$$

[ Hopf 1931 ]

# The Hopf Fibration

Right Hopf map $h\colon \mathbb{S}^3 \to \mathbb{S}^2$

The fibers $h^{-1}(p)$ for $p \in \mathbb{S}^2$ are great circles: a *Hopf bundle*

Every great circle belongs to a unique right Hopf bundle.

Isoclinic $\equiv$ belong to the same Hopf bundle
This is a transitive relation.

`http://www.geom.uiuc.edu/~banchoff/script/b3d/hypertorus.html`

# The Hopf Fibration

Right Hopf map $h\colon \mathbb{S}^3 \to \mathbb{S}^2$

The fibers $h^{-1}(p)$ for $p \in \mathbb{S}^2$ are great circles: a *Hopf bundle*

Every great circle belongs to a unique right Hopf bundle.

Isoclinic $\equiv$ belong to the same Hopf bundle
This is a transitive relation.

If all closest pairs are isoclinic
$\to$ all great circles in a connected
component of the closest-pair graph
belong to the same bundle.
$\to$ $h$ maps them to points on $\mathbb{S}^2$.

We know how to deal with $\mathbb{S}^2$!

http://www.geom.uiuc.edu/~banchoff/script/b3d/hypertorus.html

# Condensing on the 2-Sphere

Equivariant condensation on the 2-sphere:

Input: $A \subseteq \mathbb{S}^2$.

Output: $A' \subseteq \mathbb{S}^2$, $|A'| \leq \min\{|A|, 12\}$.

- $A' =$ vertices of a regular icosahedron
- $A' =$ vertices of a regular octahedron
- $A' =$ vertices of a regular tetrahedron
- $A' =$ two antipodal points, or
- $A' =$ a single point.

# Condensing on the 2-Sphere

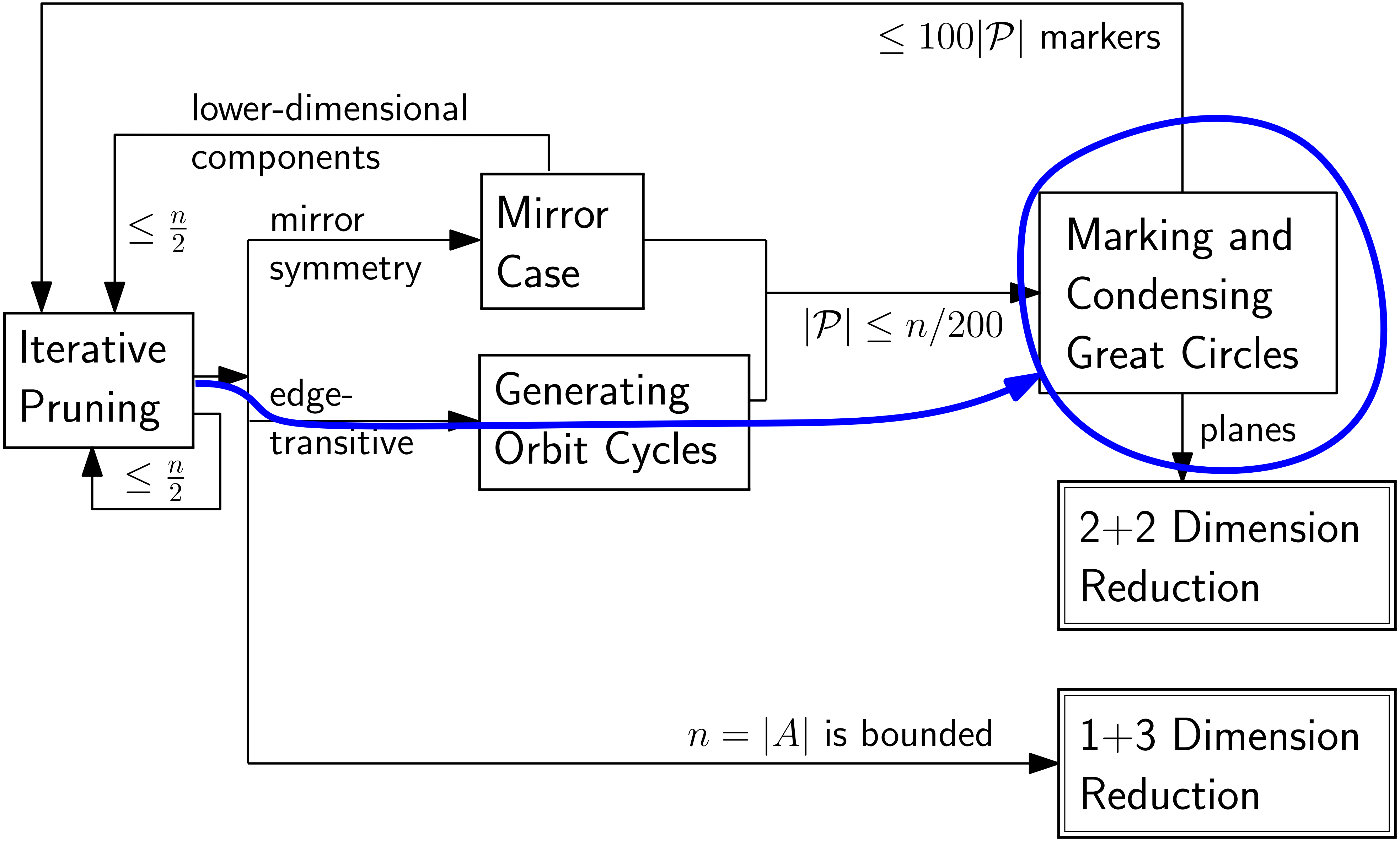
Equivariant condensation on the 2-sphere:

Input: $A \subseteq \mathbb{S}^2$.
Output: $A' \subseteq \mathbb{S}^2$, $|A'| \le \min\{|A|, 12\}$.

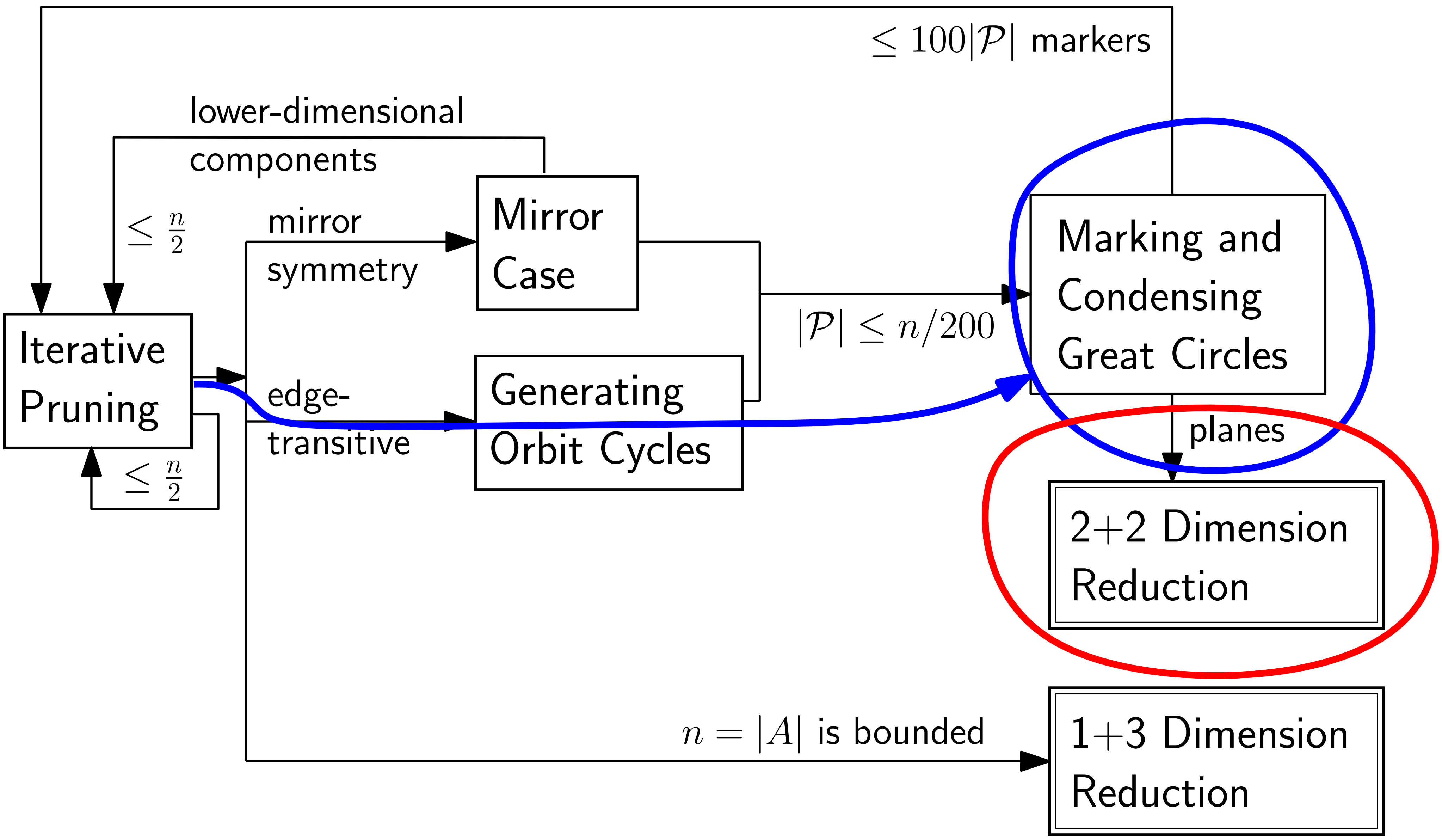
- $A'$ = vertices of a regular icosahedron
- $A'$ = vertices of a regular octahedron
- $A'$ = vertices of a regular tetrahedron
- $A'$ = two antipodal points, or
- $A'$ = a single point.

Condense each connected component of the closest-pair graph to $\le 12$ great circles.
Compute closest-pair graph (on $\mathbb{S}^5$) from scratch.
If no progress, distance between closest pairs is $\ge D_{\text{icosa}}$
$\rightarrow \le 829$ great circles $\rightarrow$ 2+2 DIMENSION REDUCTION

# Algorithm Overview



# Algorithm Overview

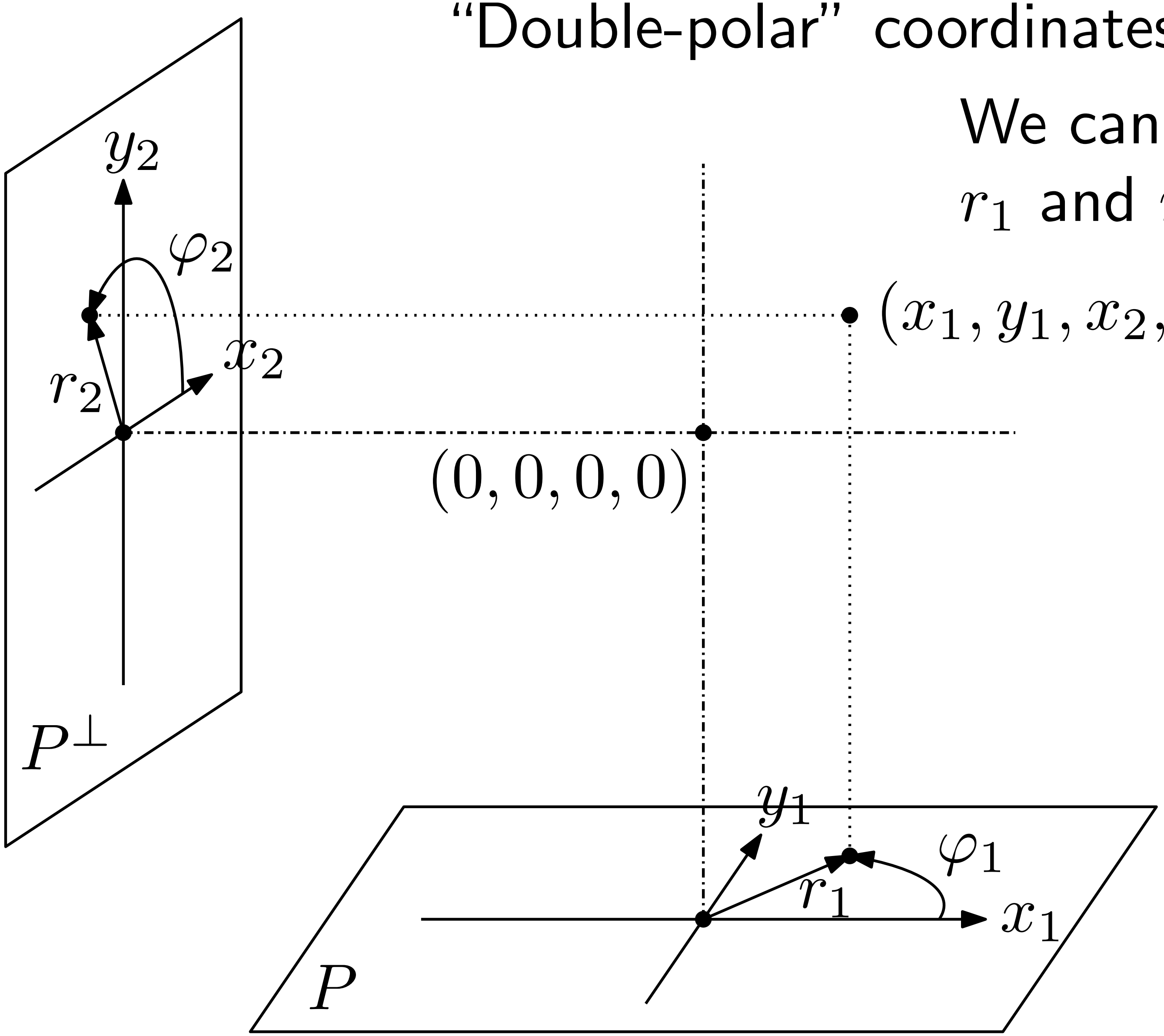


# 2+2 Dimension Reduction

We have a plane $P$ and we know its image in $B$.

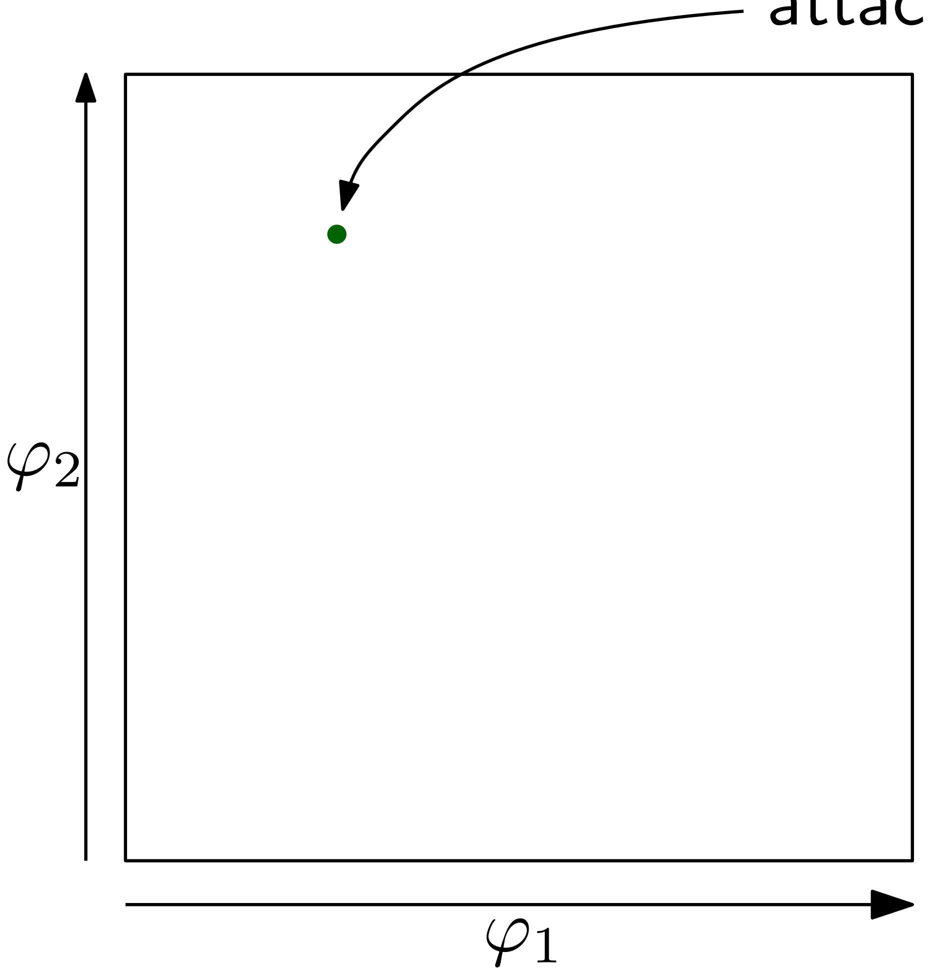
"Double-polar" coordinates $(r_1, \varphi_1, r_2, \varphi_2)$

We can change $\varphi_1$ and $\varphi_2$.
$r_1$ and $r_2$ are fixed.



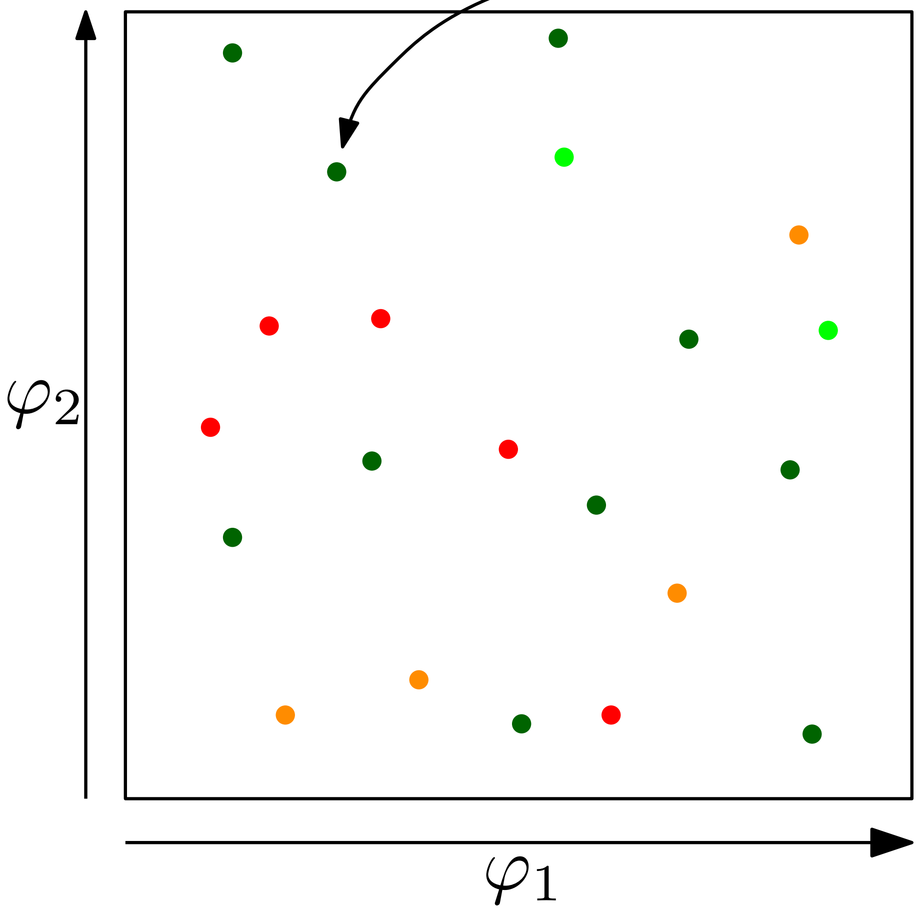
# 2+2 Dimension Reduction

attach $(r_1, r_2)$ as a label (color)



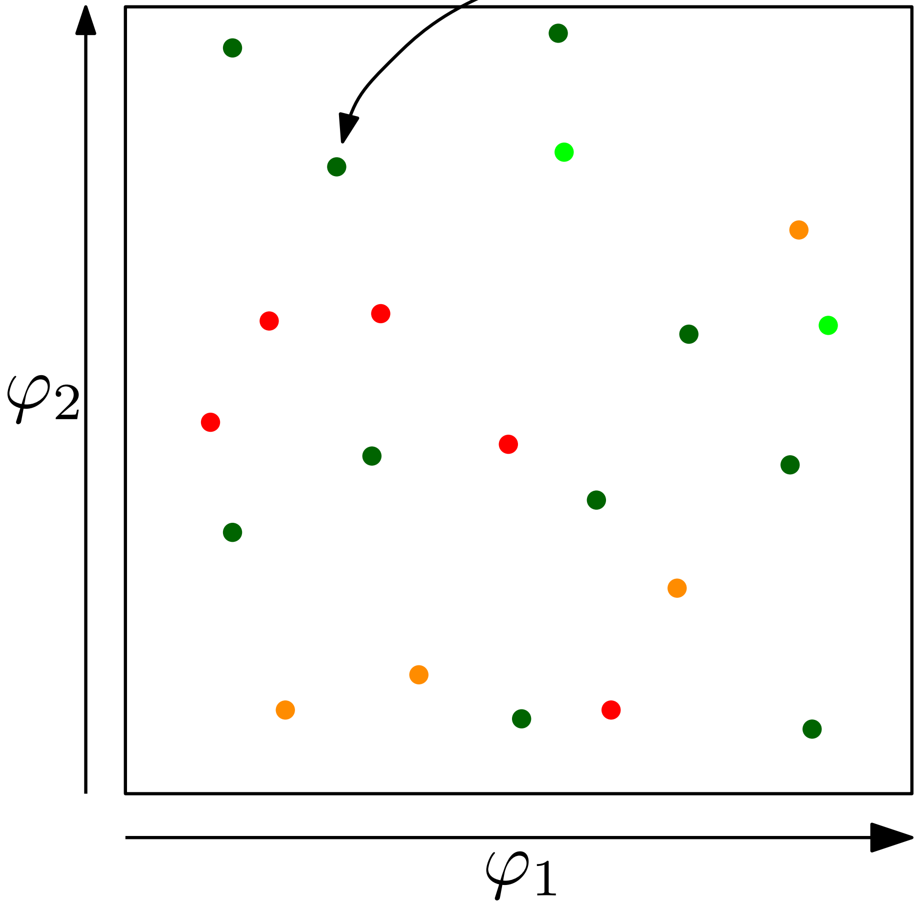
# 2+2 Dimension Reduction

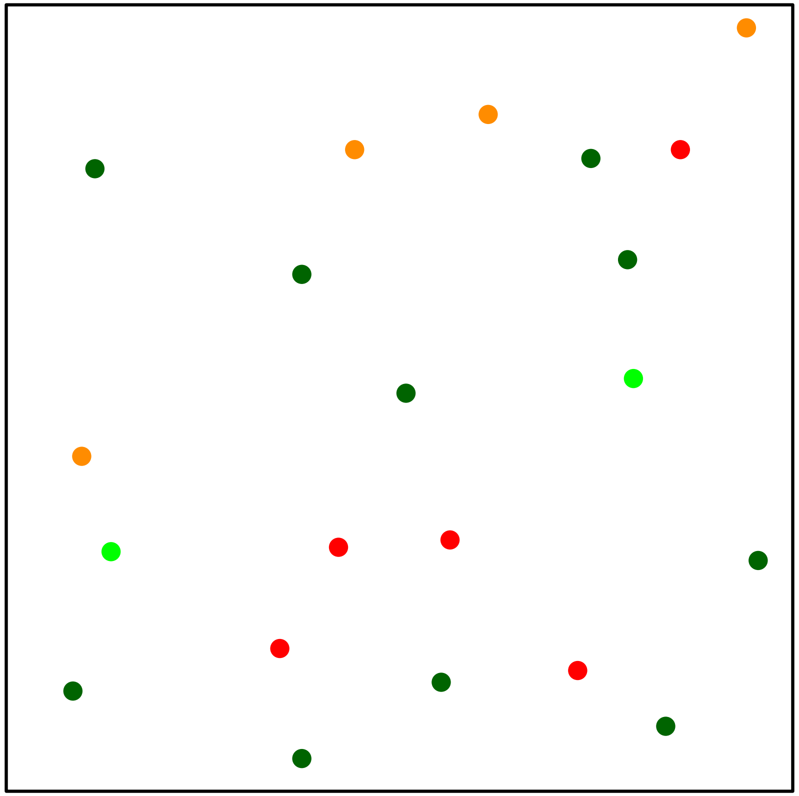
attach $(r_1, r_2)$ as a label (color)



# 2+2 Dimension Reduction

attach $(r_1, r_2)$ as a label (color)

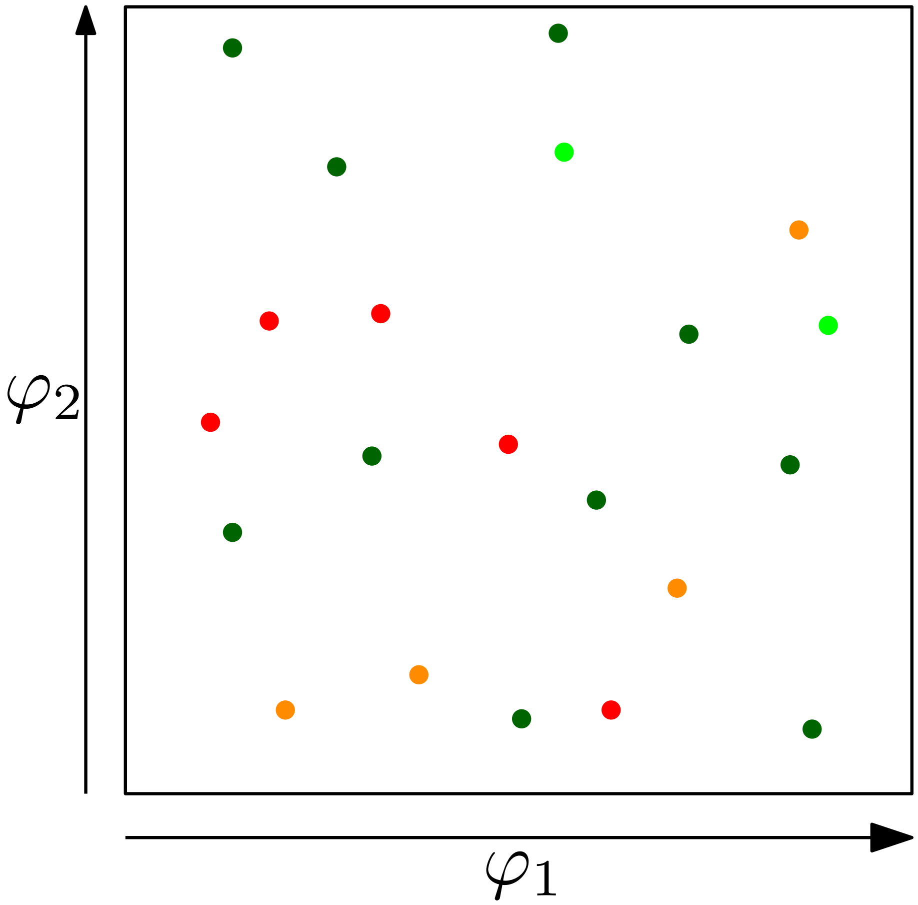




the picture for set $B$

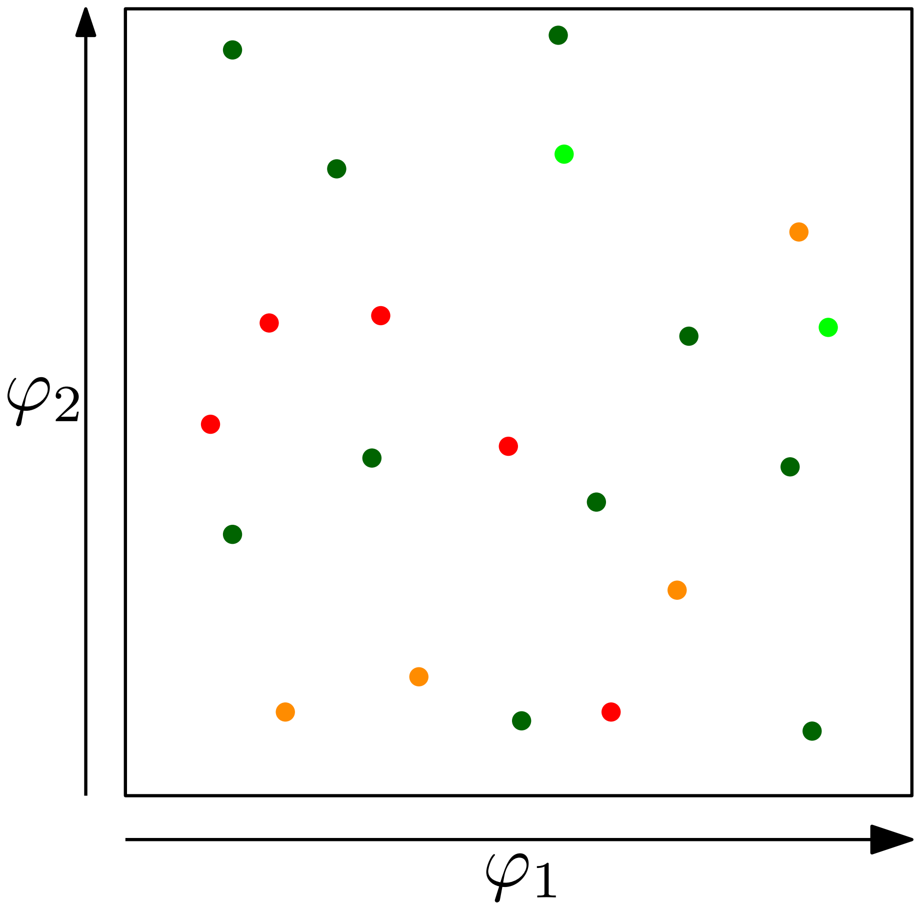
Are they the same up to translation on the $\varphi_1, \varphi_2$-torus?

# 2+2 Dimension Reduction



Prune without losing information: (CANONICAL SET)

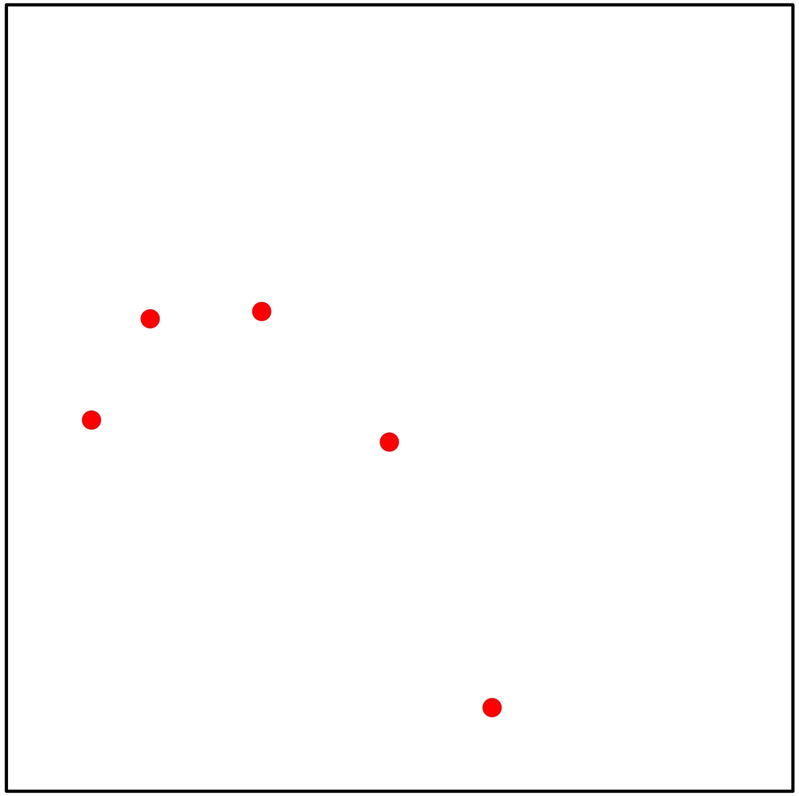
# 2+2 Dimension Reduction



Prune without losing information: (CANONICAL SET)

Pick a color class

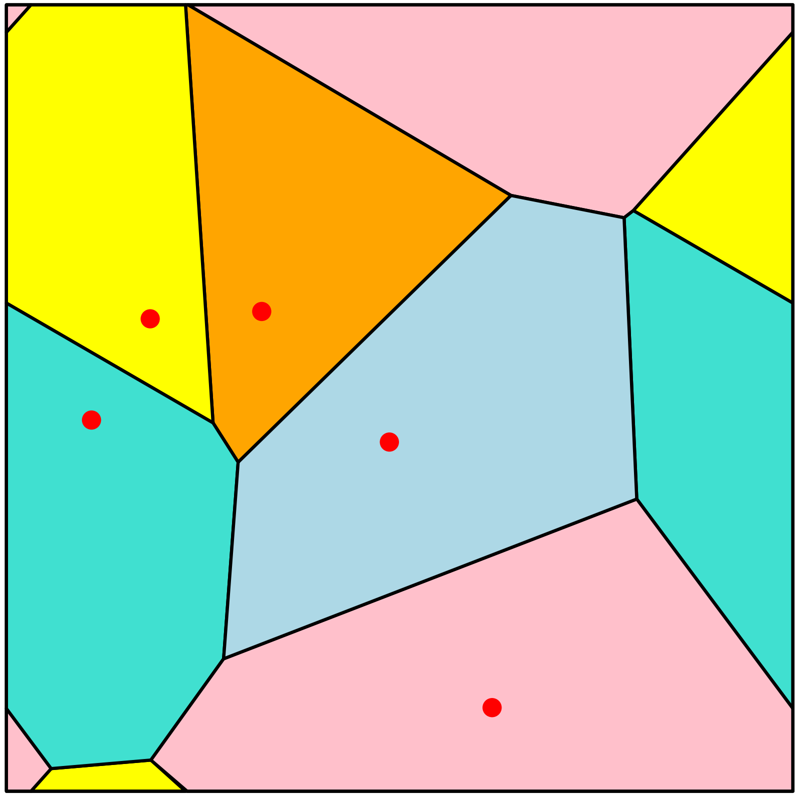
# 2+2 Dimension Reduction



Prune without losing information: (CANONICAL SET)

Pick a color class

# 2+2 Dimension Reduction

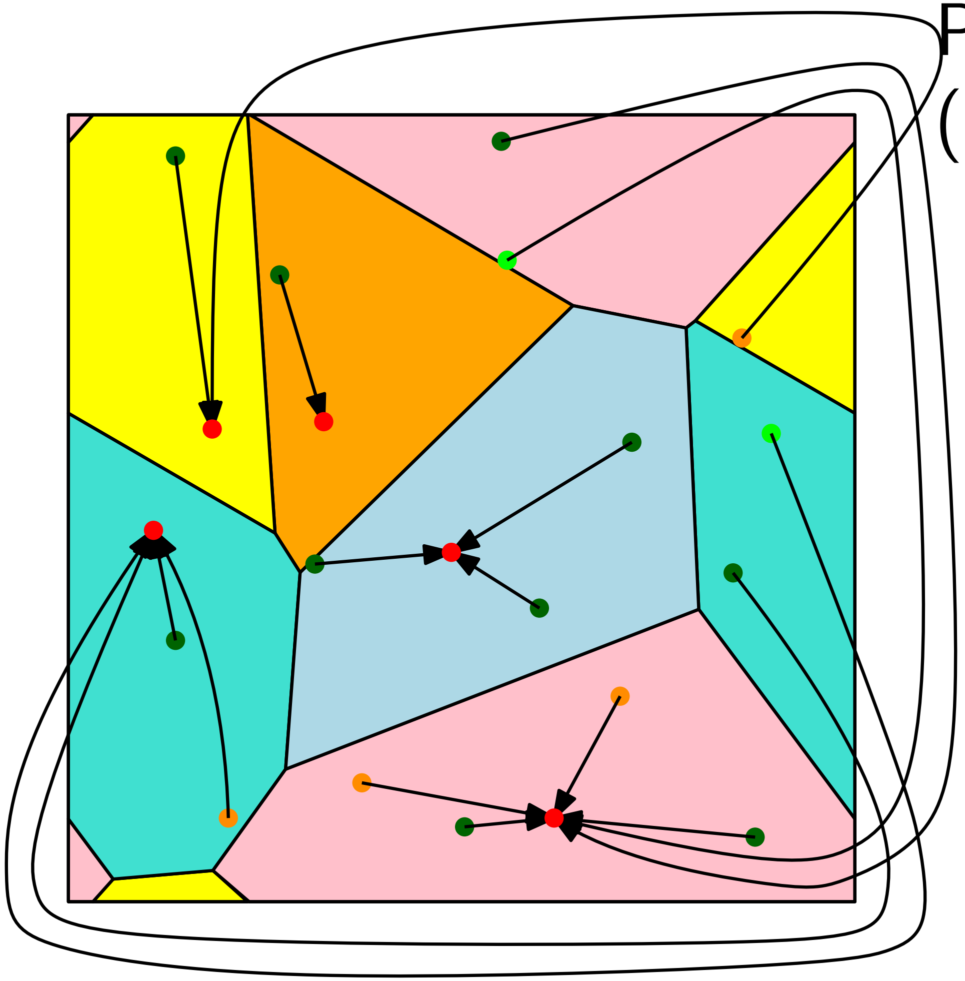


Prune without losing information: (CANONICAL SET)

Pick a color class

Compute the Voronoi diagram

# 2+2 Dimension Reduction



Prune without losing information: (CANONICAL SET)

Pick a color class

Compute the Voronoi diagram

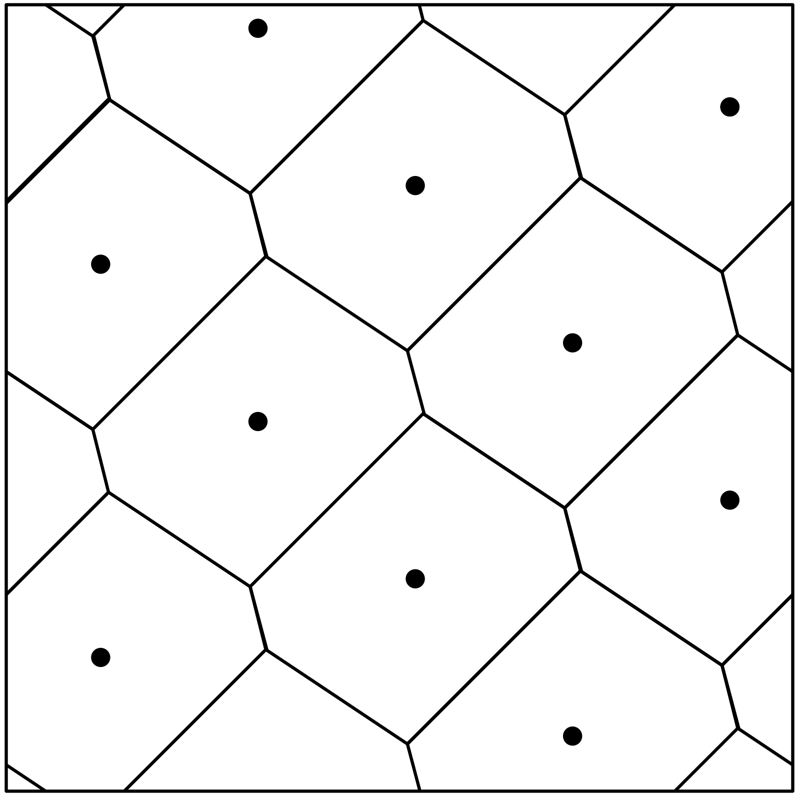
Assign other points to cells.

Refine the coloring, based on color and relative position of assigned points, shape of Voronoi cell.

Repeat.

After recoloring, the reduced set has THE SAME translational symmetries as the old set.

# 2+2 Dimension Reduction



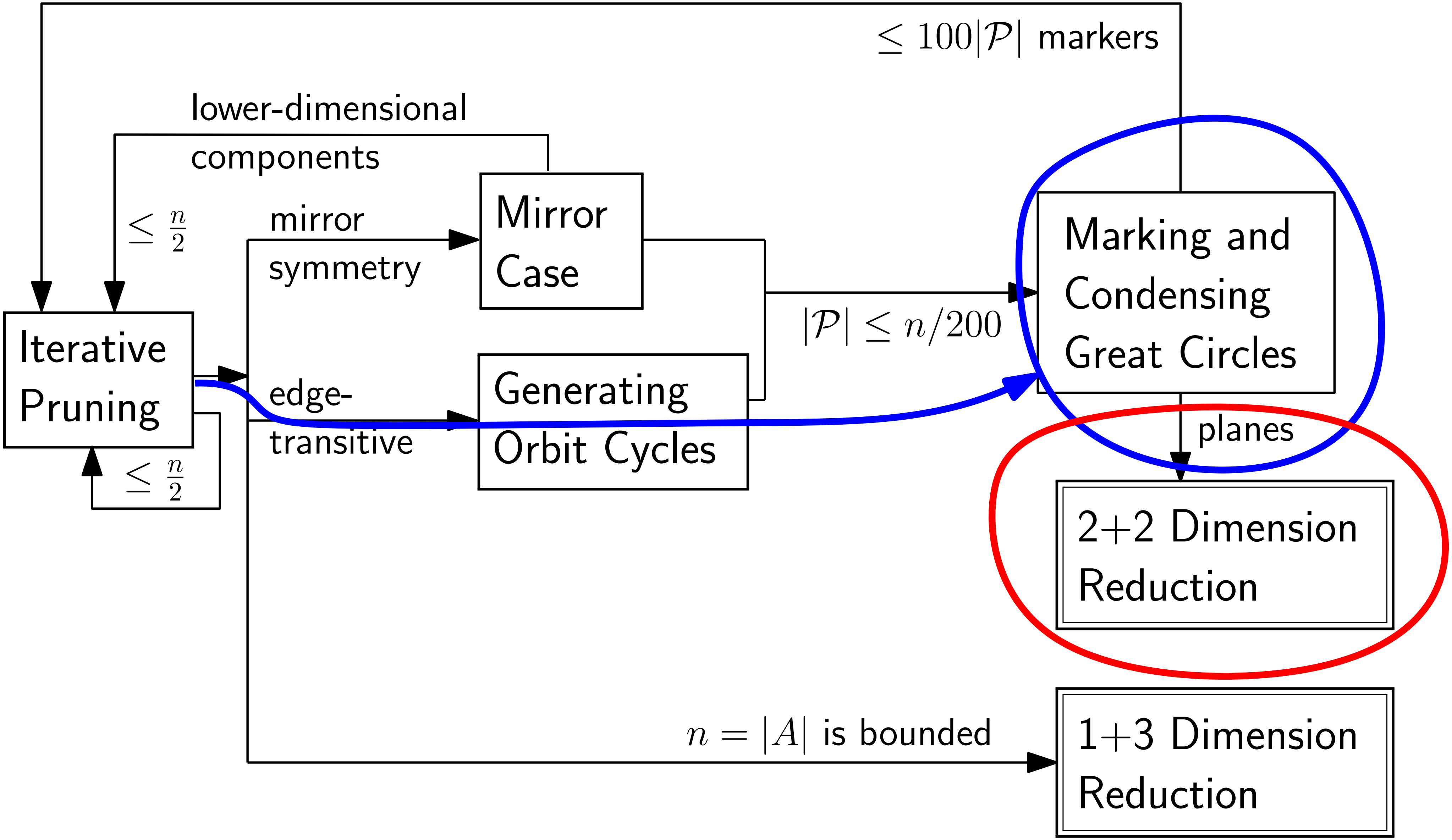
Termination:
All points have the same color and the same cell shape (a modular *lattice*)

ANY point is as good a representative as any other.

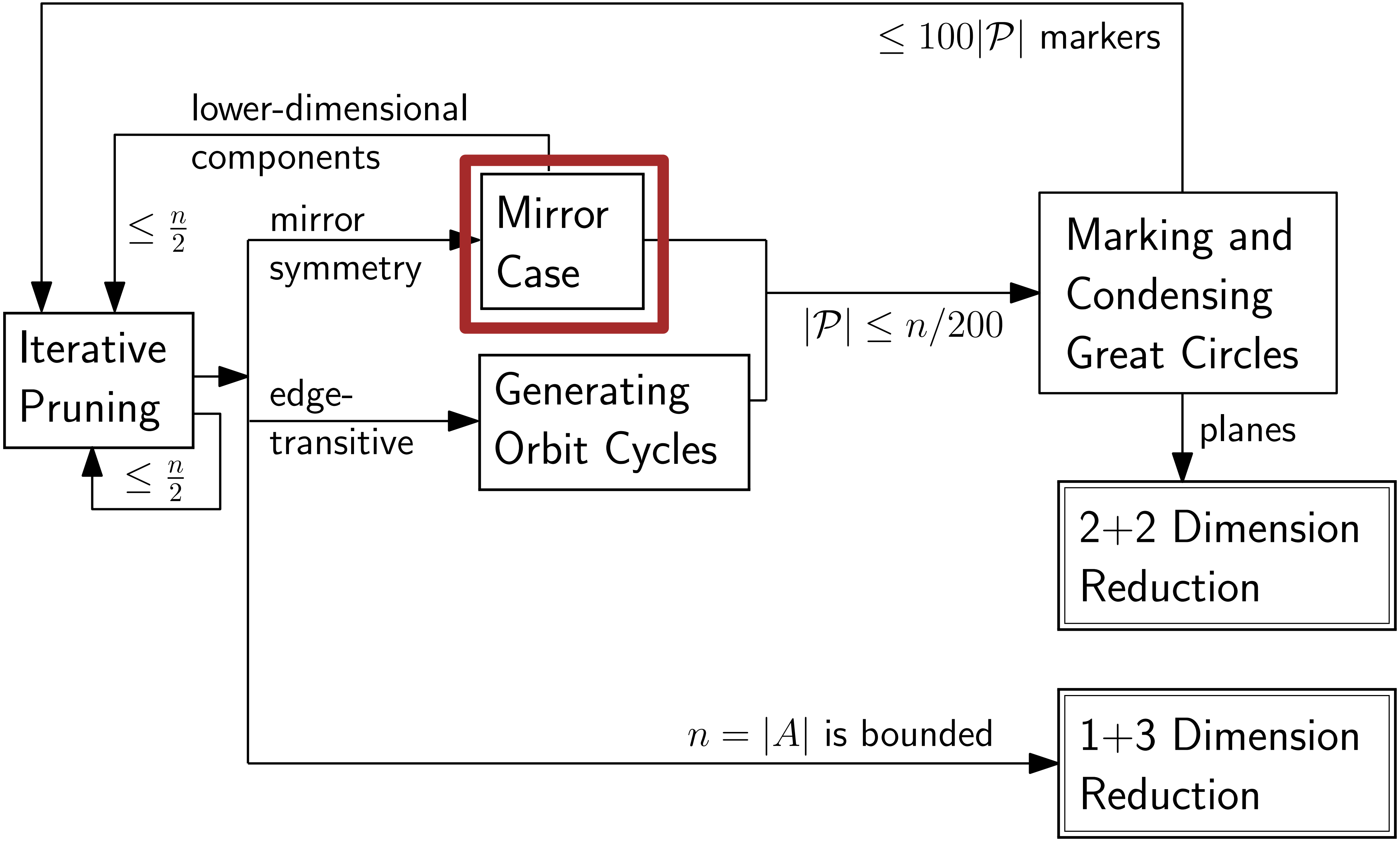
CANONICAL SET $c(A)$:
move (any) representative point to $(\varphi_1, \varphi_2) = (0, 0)$, or to $(x_1, 0, x_3, 0)$.

$$\exists T \text{ with } TP = P \text{ and } TA = B \iff c(A) = c(B)$$

# Algorithm Overview



# Algorithm Overview



# The Mirror Case

Every edge acts like a perfect mirror of the neighborhood.

$\rightarrow$ Every connnected component is the orbit of a point under a group generated by reflections.

These groups have been classified. (Coxeter groups)

- "small" components

  $\rightarrow$ pruning

- Cartesian product of 2-dimensional groups (infinite family)

  $\rightarrow$ 2+2 dimension reduction

- "large" components

  $\rightarrow |A| \le n_0$