Universal Rigidity of Generic Frameworks

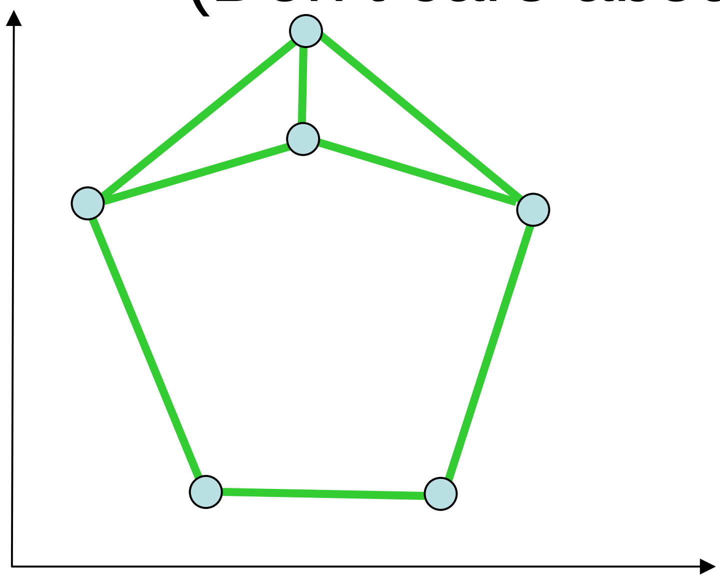
Steven Gortler

joint with Dylan Thurston

Degrees of Graph Rigidity

Framework of G

- Given a graph: "*G*",
  - v vertices, e edges
- Framework *p*: drawing in E^d
  - (Don't care about edge-edge crossing)

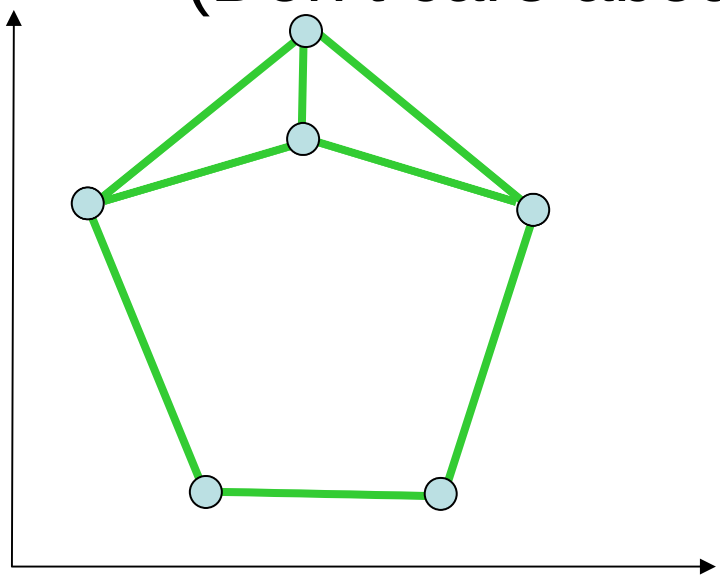


Generic

- We will restrict ourselves to frameworks that are generic in E^d
- Think “randomly perturbed” in E^d
- No algebraic coincidences
- Coincidences ruin the thms.
  - Quite typical scenario for rigidity

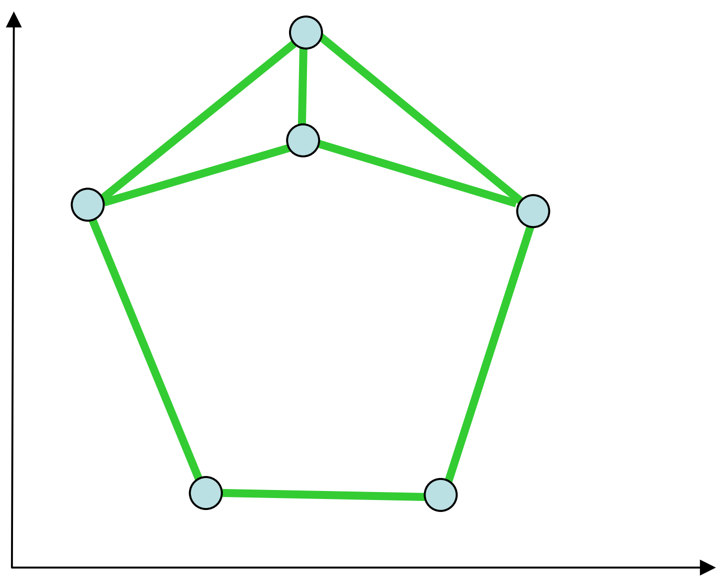
Framework of G

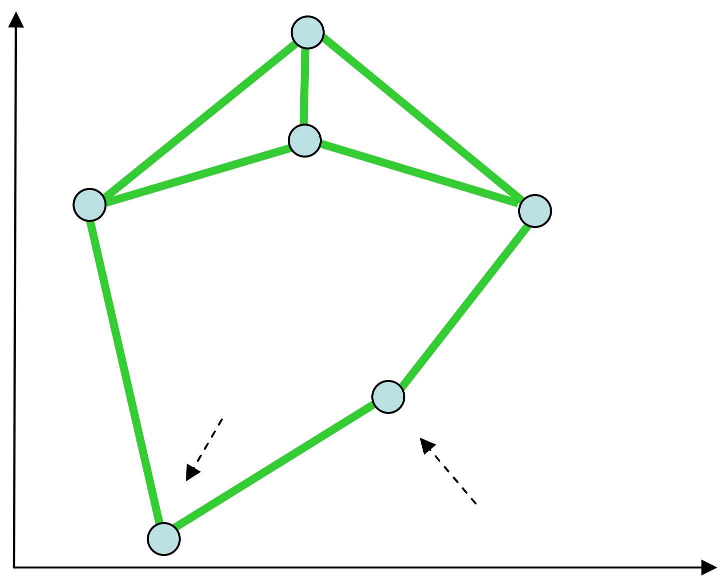
- Given a graph: "*G*",
  - v vertices, e edges
- Framework *p*: drawing in E^d
  - (Don't care about edge-edge crossing)



Locally flexible

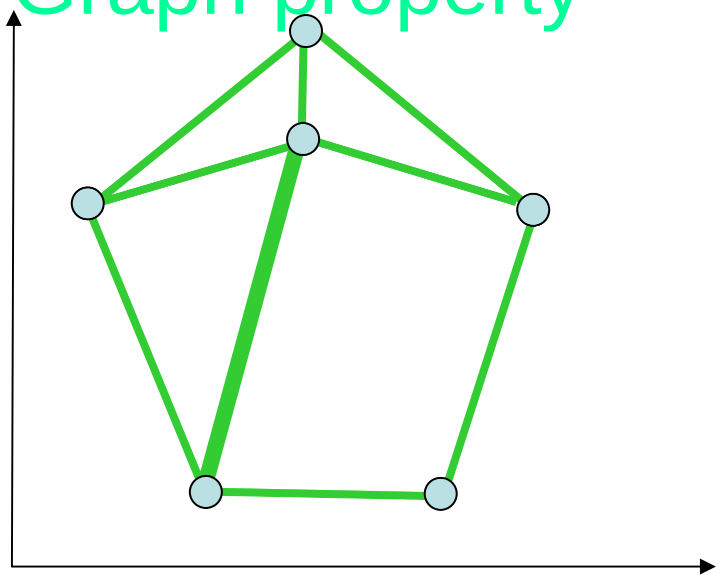
- Can find continuum of frameworks in d-dims with same edge lengths
- Graph property (for generic frameworks)





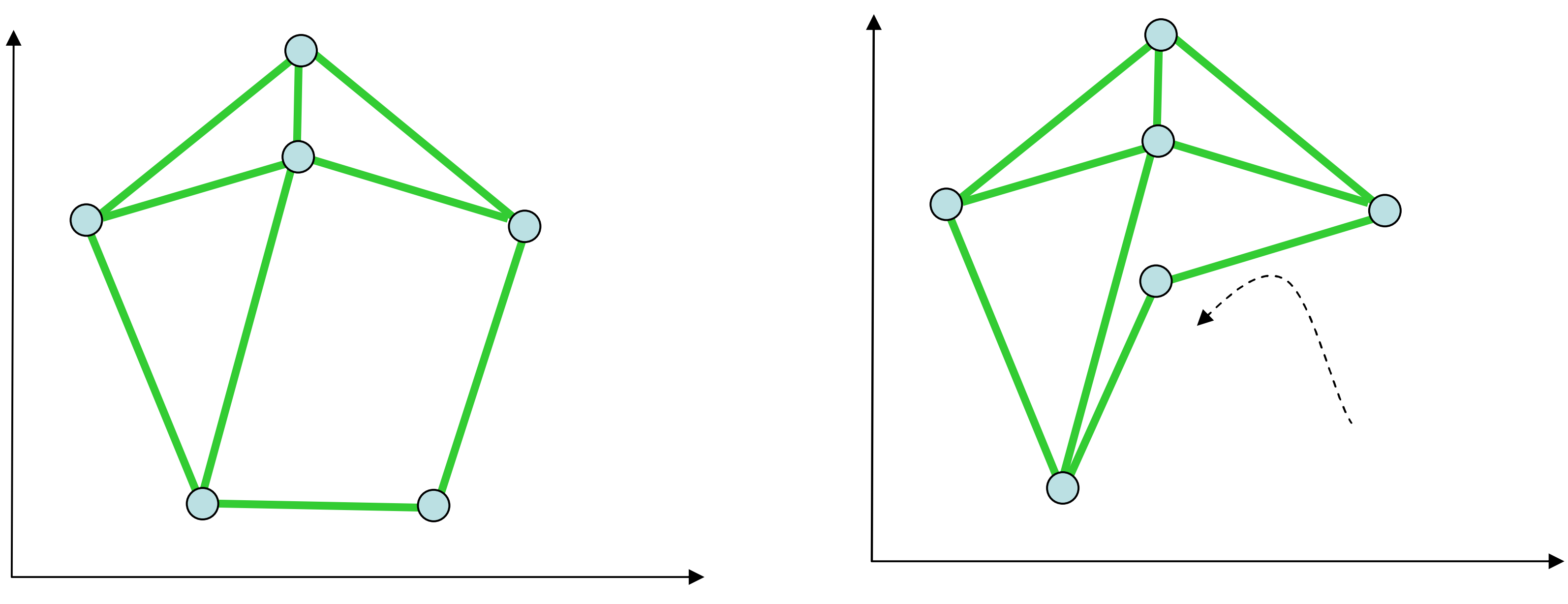
Locally rigid

- No continuum
- Well understood
  - Rank of appropriate matrix
- Graph property



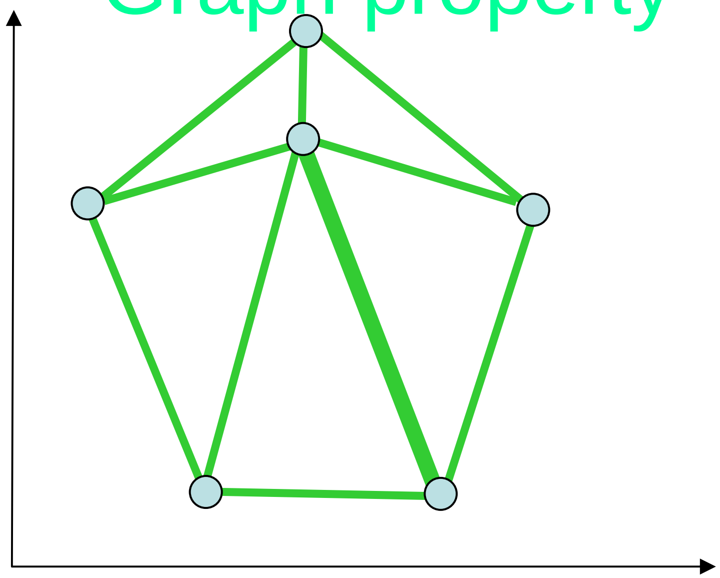
Locally rigid

- No continuum
- But can find other discrete framework in d-dims



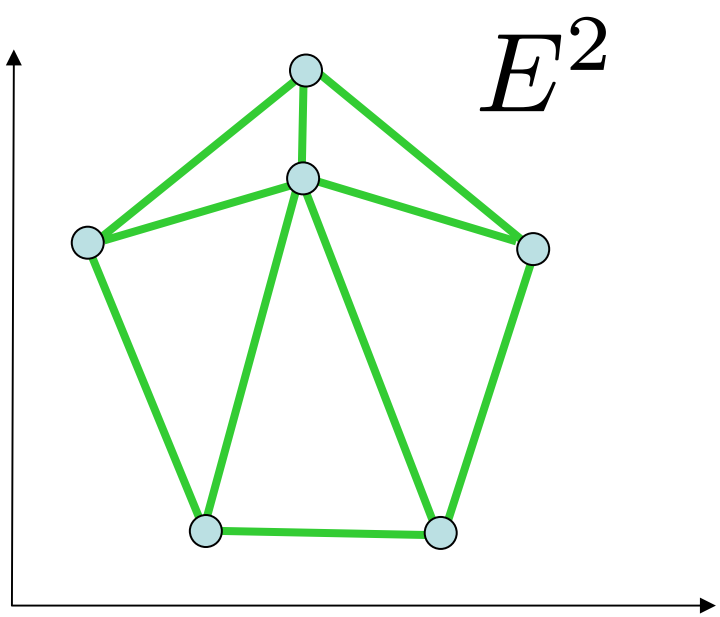
Globally Rigid

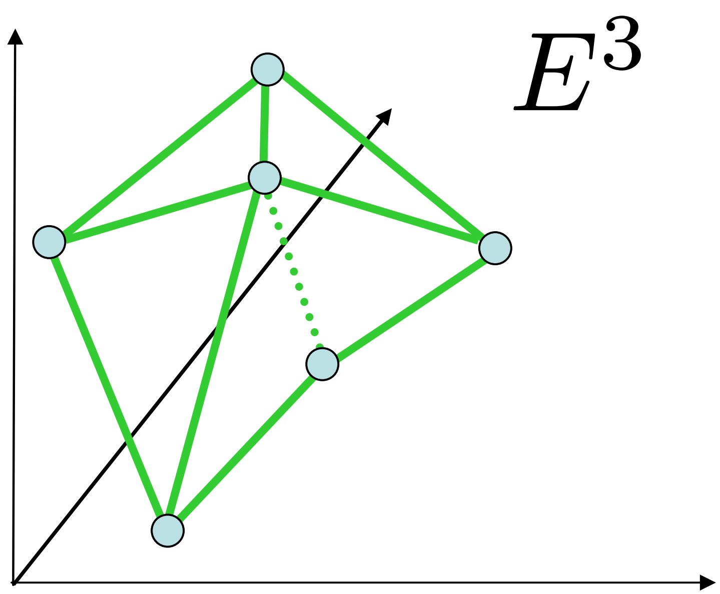
- No other framework in d-dims
- Well understood [C'82, GHT '07]
  - Rank of appropriate matrix
- Graph property



Globally Rigid

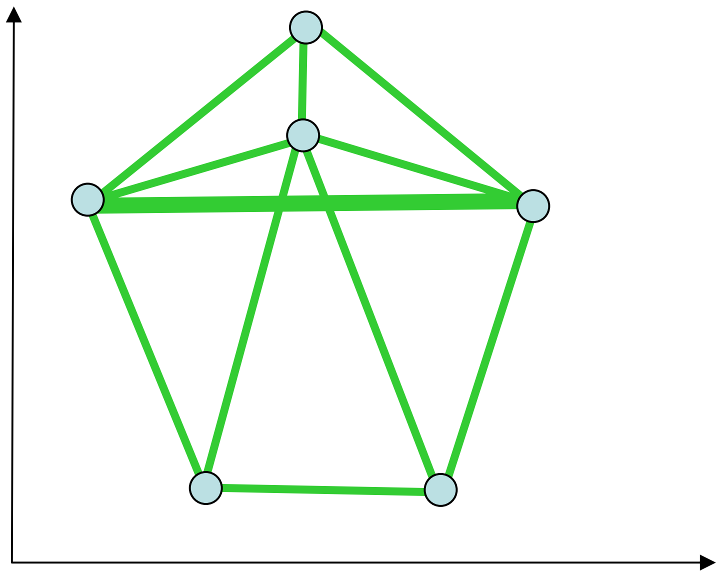
- No other framework in d-dims
- But can be other frameworks in higher dims





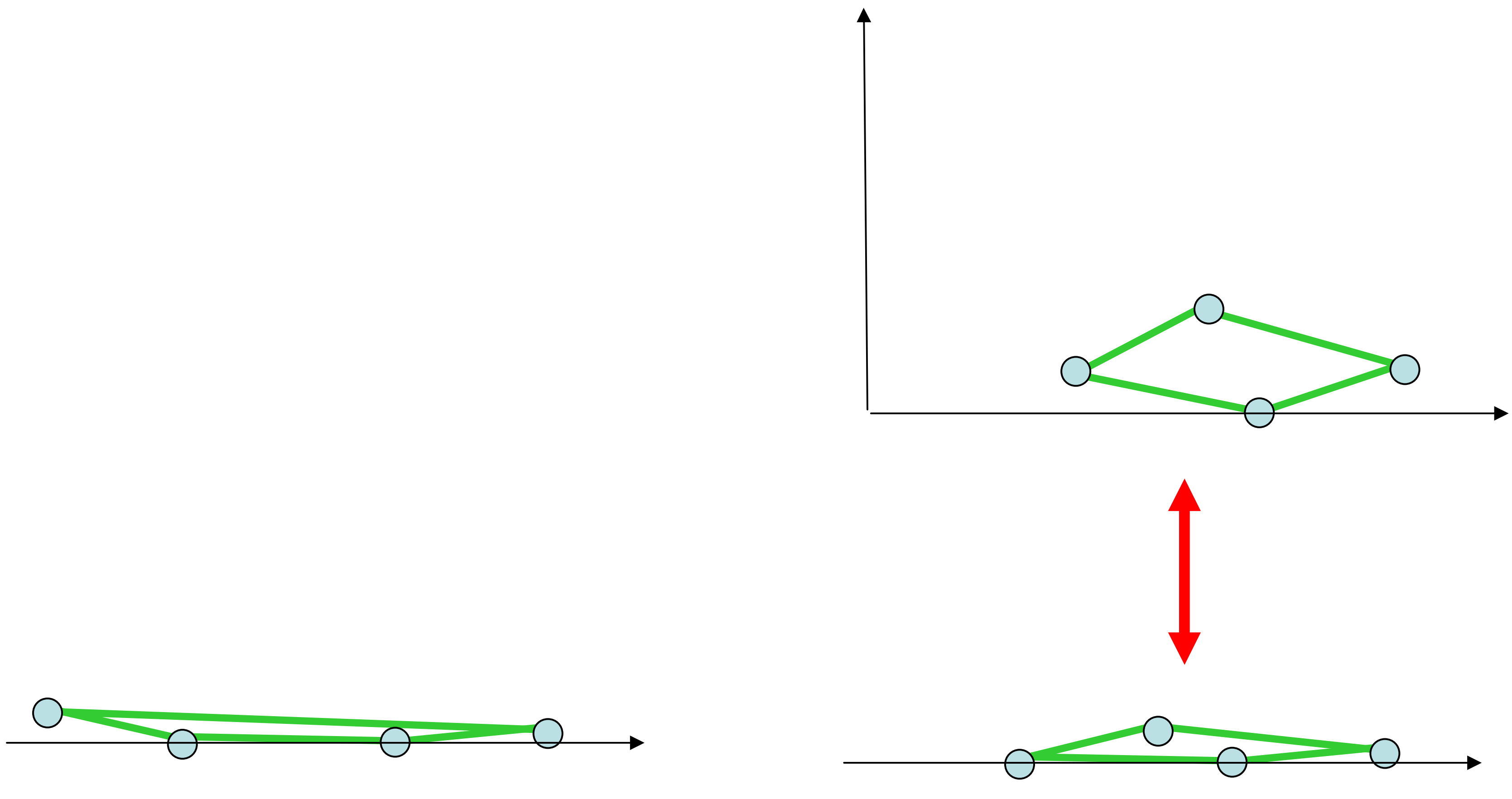
Universally Rigid (UR)

- No other framework in **ANY**-dims

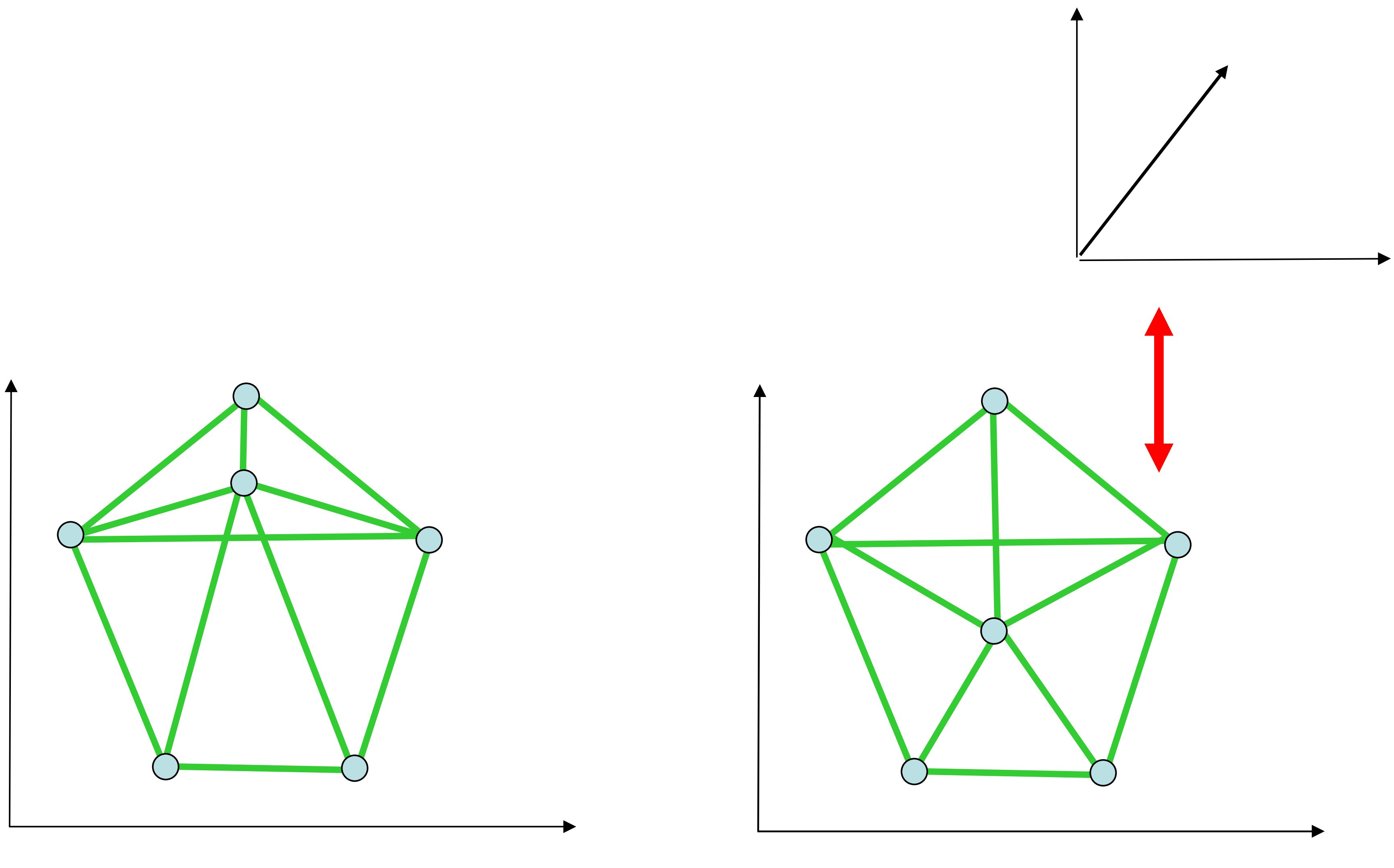


UR: can depend on the generic embedding: 1D

- Unlike local and global rigidity

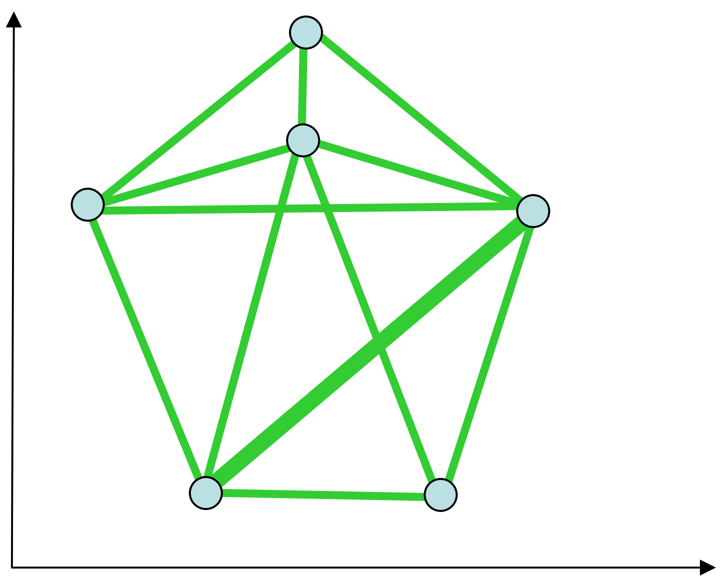


UR: can depend on the generic embedding: 2D

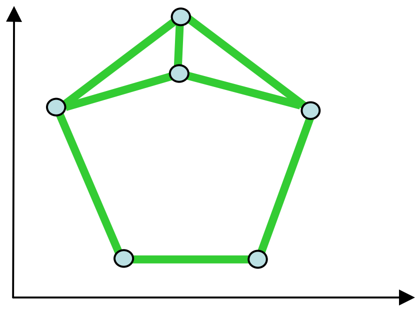


Genericaly UR Graphs

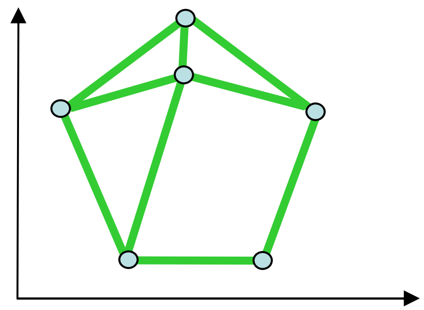
- Graphs such that ALL d-dim generic frameworks are UR



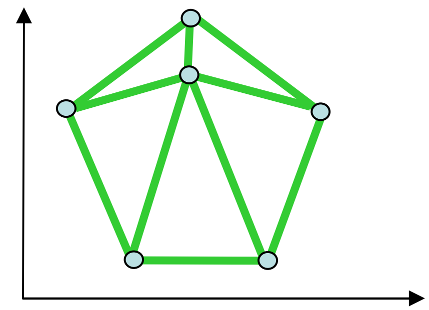
Degrees of rigidity



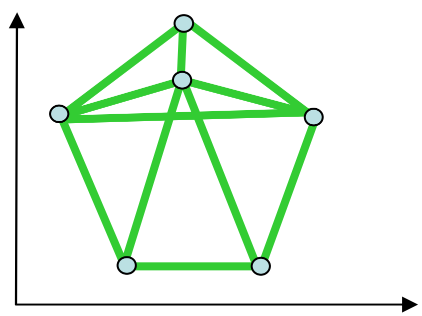
Locally Flexible graph



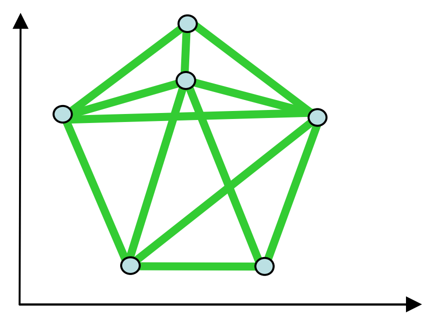
Locally Rigid graph



Globally Rigid graph

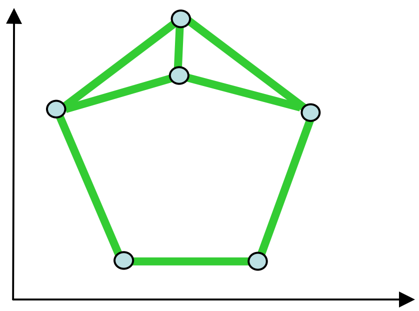


Universally Rigid Framework

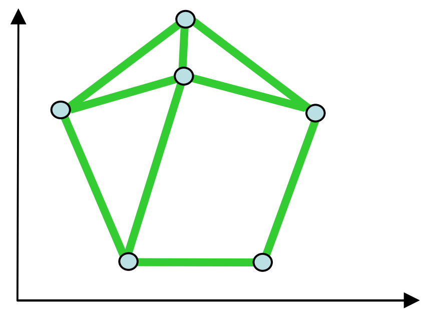


Universally Rigid Graph

Degrees of rigidity

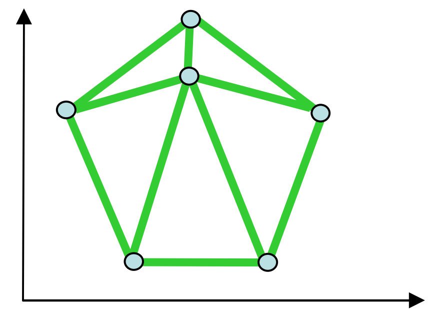


Locally Flexible graph

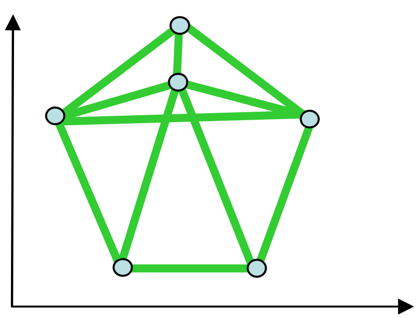


Locally Rigid graph

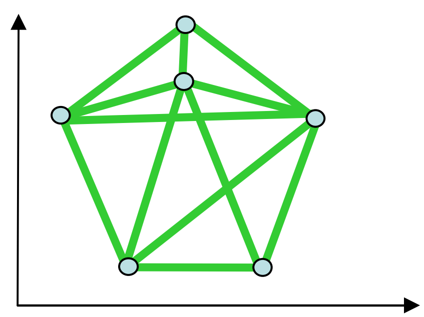
2007



Globally Rigid graph

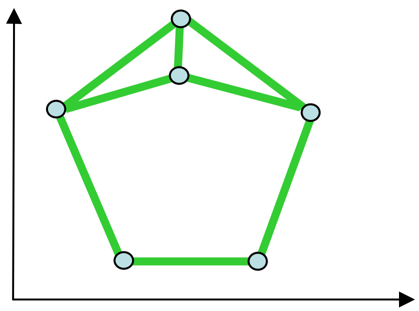


Universally Rigid Framework

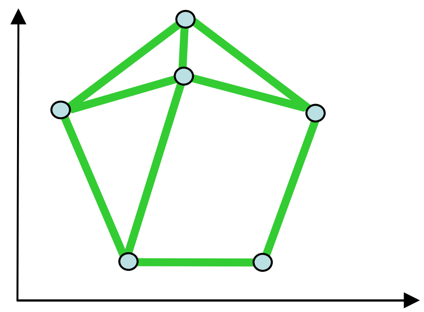


Universally Rigid Graph

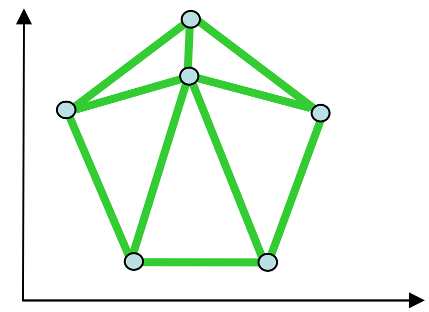
Degrees of rigidity



Locally Flexible graph

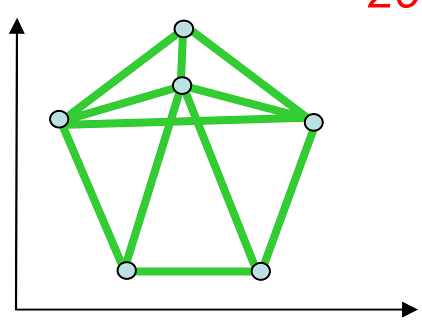


Locally Rigid graph

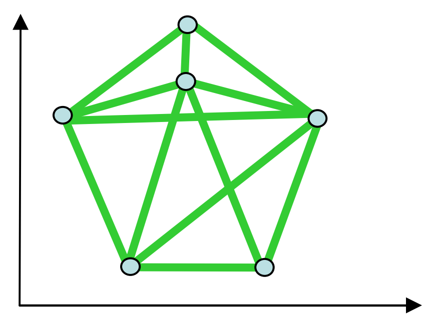


Globally Rigid graph

2009



Universally Rigid Framework

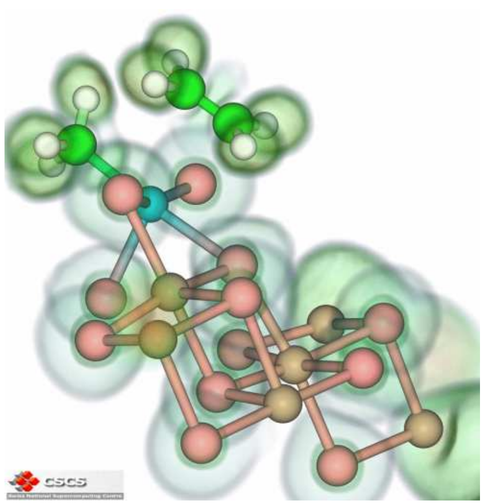


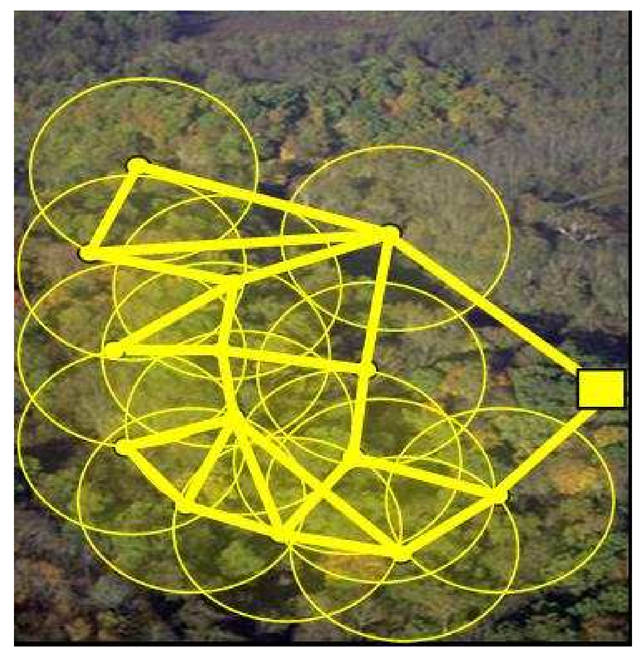
Universally Rigid Graph

Motivation for UR

Motivation for UR

- General embeddings from distances is NP-hard
- But it is useful in application
  – Molecular structure from NMR data
  – Sensor networks





Motivation for UR

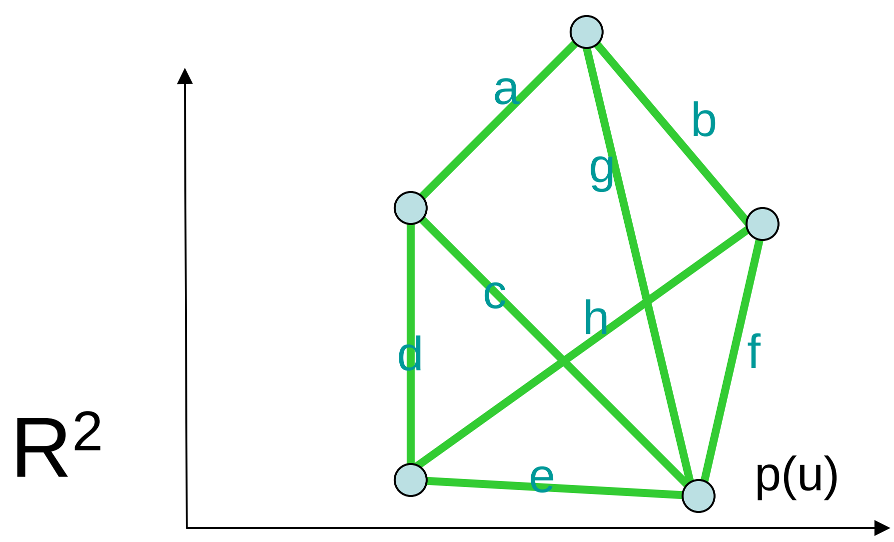
- Try semidefinite programming (SDP) [LLR ’95]
  - Unknown Gram matrix
    - Positive semidefinite (PSD) constraint
  - Length squared constraints are linear
- Will get solution in very high dimension
- Rank constraints are non-convex
  - UR: No need for explicit rank constraint
    - [So Ye ‘07]
- Wish to characterize this class

Define terms

Equilibrium Stress Vector (ESV) of p

- A real number w_{uv} on each edge e_{uv}

$$\forall u \sum_v \omega_{uv}[\rho(v) - \rho(u)] = \vec{0}$$

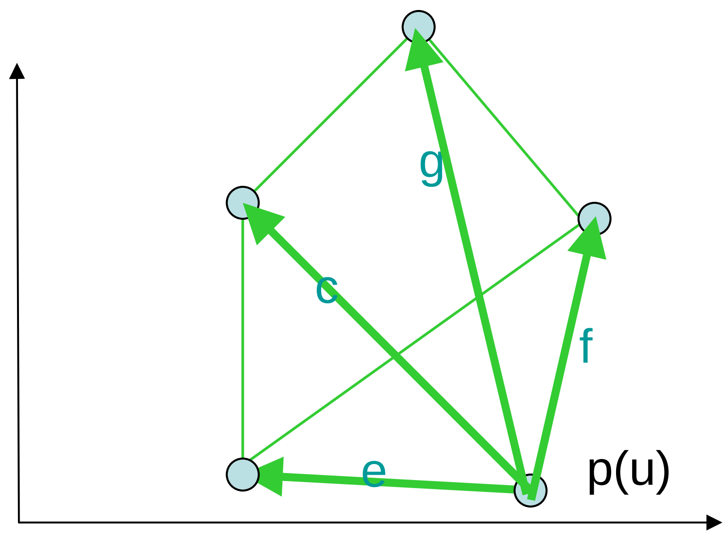


ESV of p

- A real number w_{uv} on each edge e_{uv}

$$\forall u \sum_v \omega_{uv}[\rho(v) - \rho(u)] = \vec{0}$$

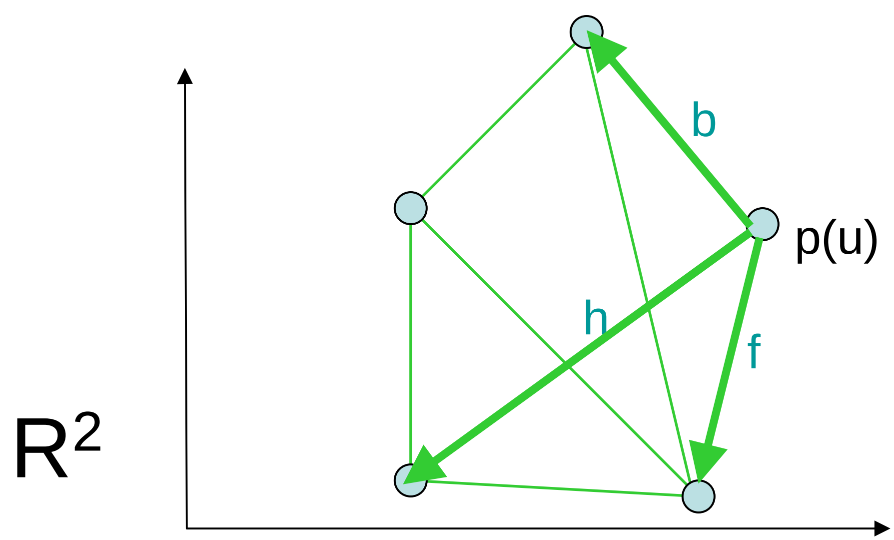
R^2



ESV of p

- A real number w_{uv} on each edge e_{uv}

$$\forall u \sum_v \omega_{uv}[\rho(v) - \rho(u)] = \vec{0}$$

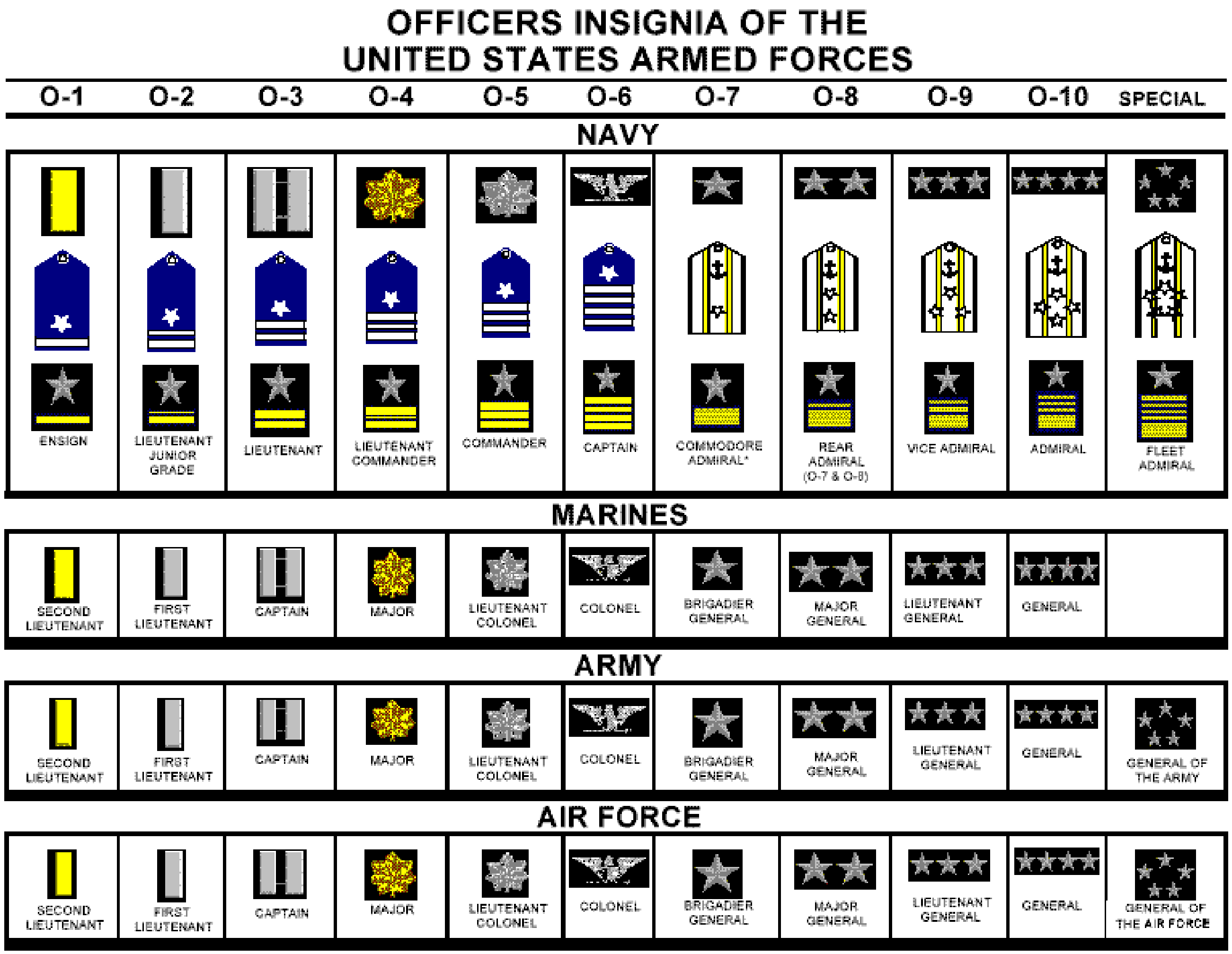


Equilibrium Stress Matrix (ESM) of p

- v by v matrix
- $\Omega_{uv} := \Omega_{vu} := \omega_{uv}$
- $\Omega_{uv} := 0 \text{ for non-edges}$
- $\Omega_{uu} := -\sum_v \omega_{uv} \text{ for diagonals}$

- "Matricize the ESV"

Rank


OFFICERS INSIGNIA OF THE
UNITED STATES ARMED FORCES
O-1
O-2
O-3
O-4
O-5
O-6
O-7
O-8
O-9
O-10
SPECIAL
NAVY
ENSIGN
LIEUTENANT JUNIOR GRADE
LIEUTENANT
LIEUTENANT COMMANDER
COMMANDER
CAPTAIN
COMMODORE ADMIRAL*
REAR ADMIRAL (O-7 & O-8)
VICE ADMIRAL
ADMIRAL
FLEET ADMIRAL
MARINES
SECOND LIEUTENANT
FIRST LIEUTENANT
CAPTAIN
MAJOR
LIEUTENANT COLONEL
COLONEL
BRIGADIER GENERAL
MAJOR GENERAL
LIEUTENANT GENERAL
GENERAL
ARMY
SECOND LIEUTENANT
FIRST LIEUTENANT
CAPTAIN
MAJOR
LIEUTENANT COLONEL
COLONEL
BRIGADIER GENERAL
MAJOR GENERAL
LIEUTENANT GENERAL
GENERAL
GENERAL OF THE ARMY
AIR FORCE
SECOND LIEUTENANT
FIRST LIEUTENANT
CAPTAIN
MAJOR
LIEUTENANT COLONEL
COLONEL
BRIGADIER GENERAL
MAJOR GENERAL
LIEUTENANT GENERAL
GENERAL
GENERAL OF THE AIR FORCE

Equilibrium Stress Matrix (ESM) of p

- By construction, kernel must contain:
  - Each of the d-coordinates of p

$$\Omega \rho^j$$

Equilibrium Stress Matrix (ESM) of p

- By construction, kernel must contain:
  - Each of the d-coordinates of p

$$\Omega\rho^j = \sum_v \omega_{uv}[\rho^j(v) - \rho^j(u)]$$

Equilibrium Stress Matrix (ESM) of p

- By construction, kernel must contain:
  - Each of the d-coordinates of p

$$\Omega\rho^j = \sum_v \omega_{uv}[\rho^j(v) - \rho^j(u)] = 0$$

Equilibrium Stress Matrix (ESM) of p

- By construction, kernel must contain:
  - Each of the d-coordinates of p
  - The all-ones vector

$$\Omega\rho^j = \sum_v \omega_{uv}[\rho^j(v) - \rho^j(u)] = 0$$
$$\Omega\vec{1} = \sum_v \omega_{uv}[1 - 1] = 0$$

Equilibrium Stress Matrix (ESM) of p

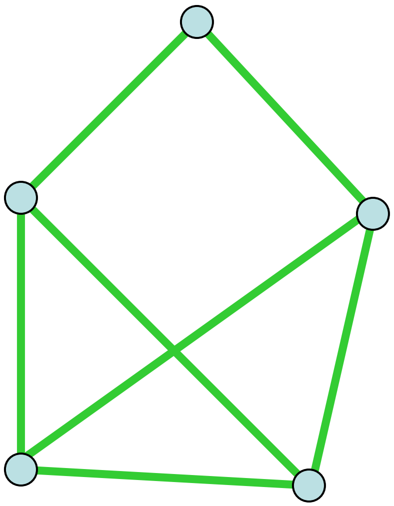
- By construction, kernel must contain:
  - Each of the d-coordinates of p
  - The all-ones vector
- Maximal possible rank is v-d-1

$$\Omega \rho^j = \sum_v \omega_{uv} [\rho^j(v) - \rho^j(u)] = 0$$

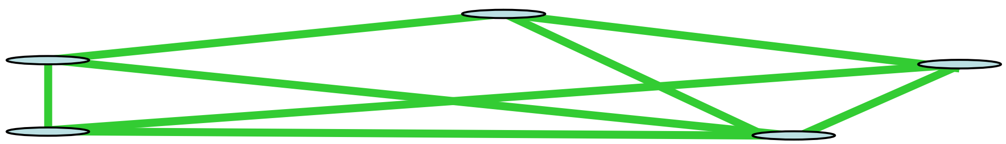
$$\Omega \vec{1} = \sum_v \omega_{uv} [1 - 1] = 0$$

Maximal possible rank is v-d-1

- Any affine transform 'T(*p)*' will satisfy **any** stress w of p
- If Ω has maximal rank, then these are the only satisfying drawings



p



T(p)

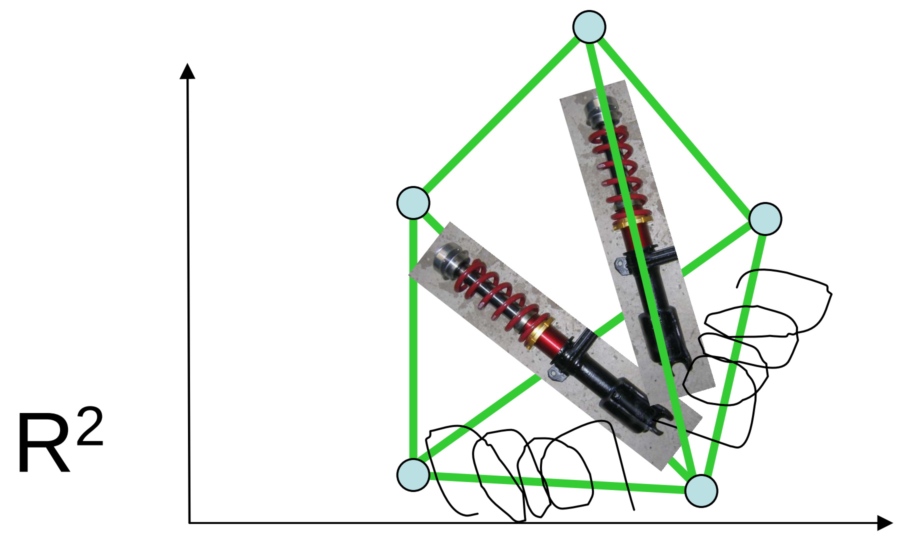
Springs and Struts

Stress Energy

- Stress vector -> a quadratic spring/strut energy

$$E(\rho) := \sum_{uv} \omega_{uv} |\rho(v) - \rho(u)|^2$$

- ESV condition == p is an equilibrium point of this energy
  - Max/Min/Saddle

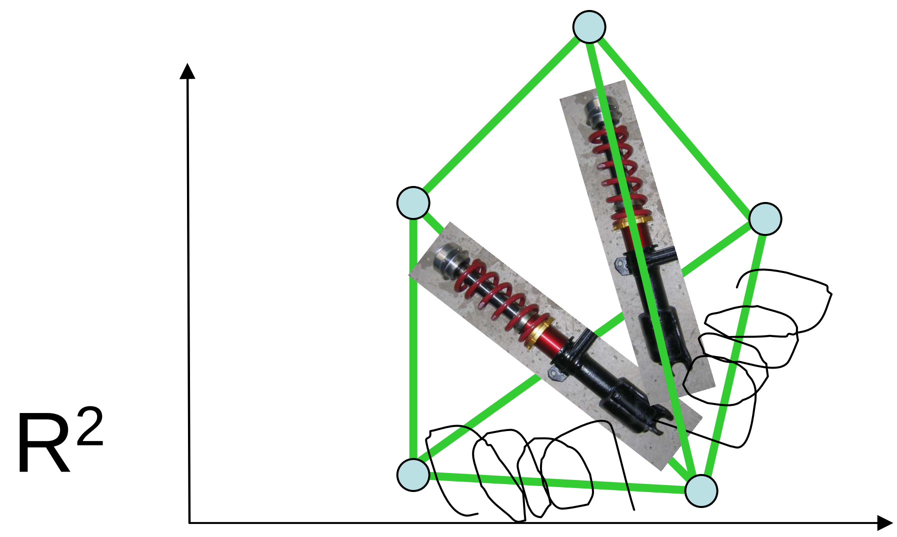


Stress Energy

- Stress vector -> a quadratic spring/strut energy

$$E(\rho) := \sum_{uv} \omega_{uv} |\rho(v) - \rho(u)|^2 = \sum_j (\rho^j)^t \Omega \rho^j$$

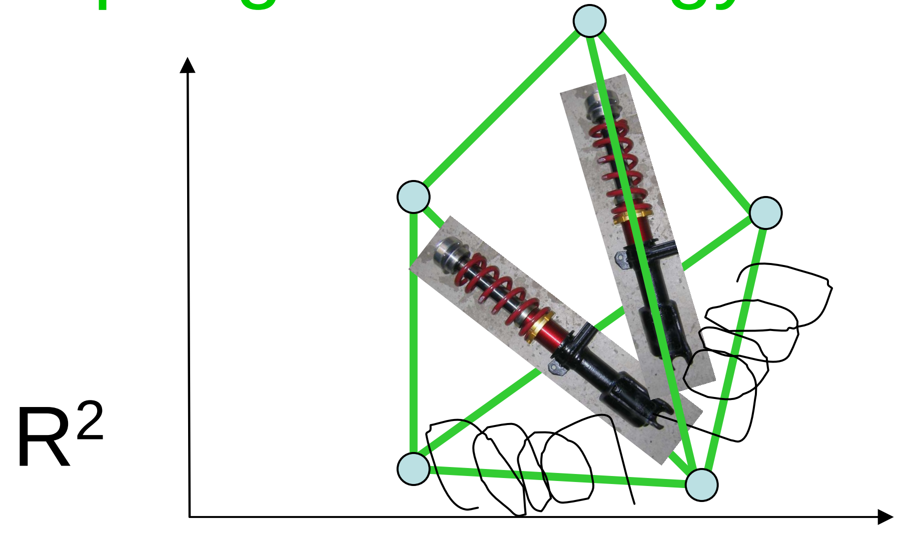
- ESV condition == p is an equilibrium point of this energy
  - Max/Min/Saddle



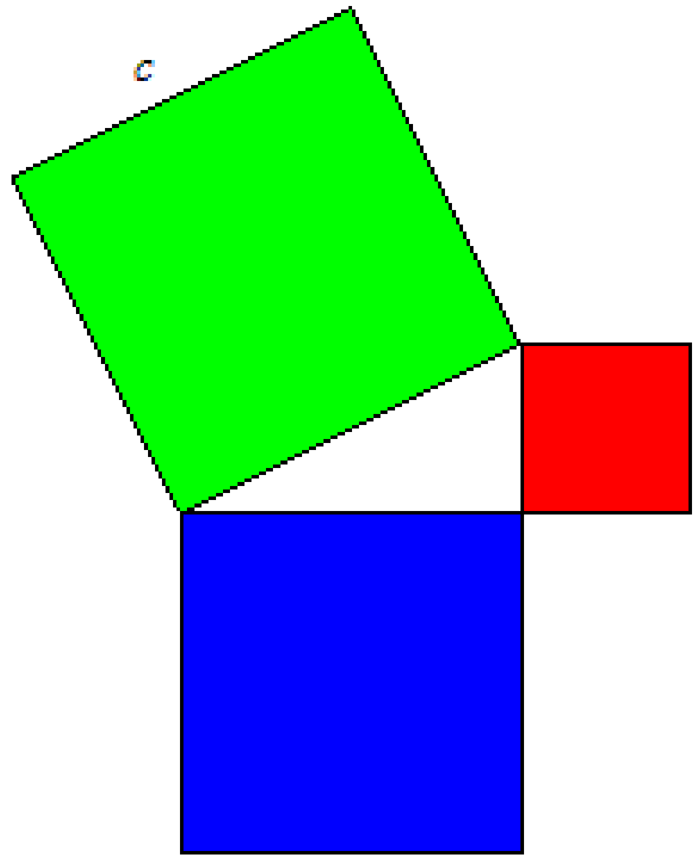
Positive semidefinite

$$E(\rho) := \sum_j (\rho^j)^t \Omega \rho^j$$

- If Ω is positive semidefinite, then p is in fact a min of the spring/strut energy



Theorems



Sufficiency [Con 82]

If p is generic in E^d

p has a ESM: Ω of rank v-d-1.

Ω is PSD

Then

p is UR

Main result today [GT 09]

If p is generic in E^d

p is UR

Then

p has a ESM: Ω of rank v-d-1.

Ω is PSD

Characterization

If p is generic in E^d

p is UR

==

p has a ESM: Ω of rank v-d-1.

Ω is PSD

Compare to GGR [Con 89+ GHT 07]

If p is generic in E^d

p is UR GR in E^d

==

p has a ESM: Ω of rank v-d-1.

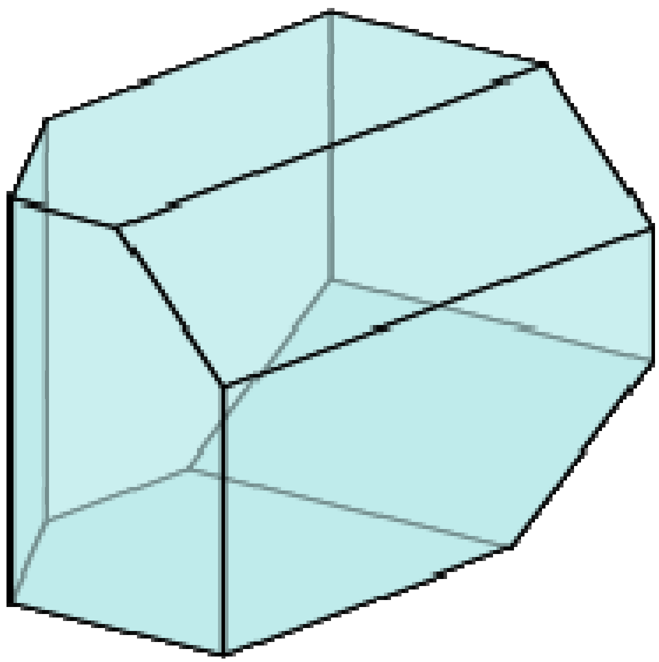
Ω is PSD

Proof


Austin Nichols
WILD TURKEY
KENTUCKY STRAIGHT
BOURBON
WHISKEY
Real Kentucky

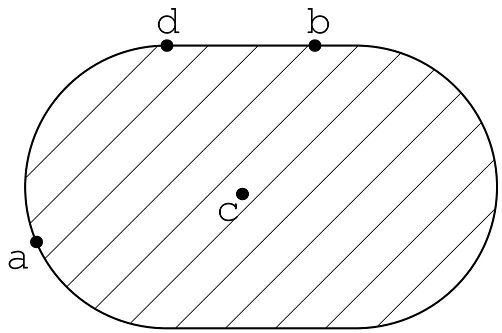
Convex preliminaries

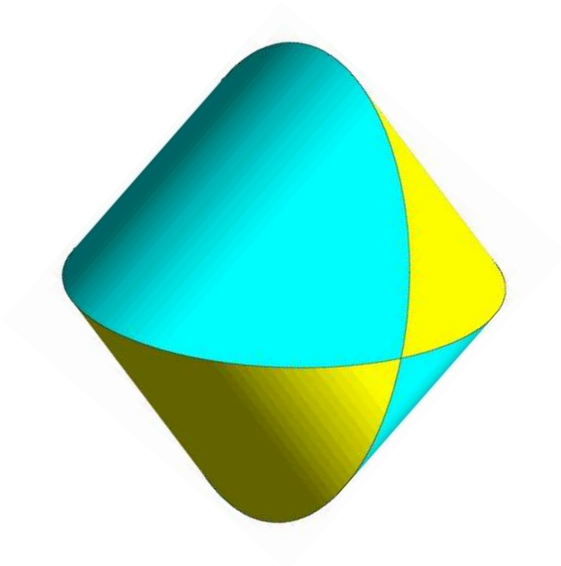
- Stratify by “dimension”
- Obvious for polytopes
- Two notions in general



Convex extremity

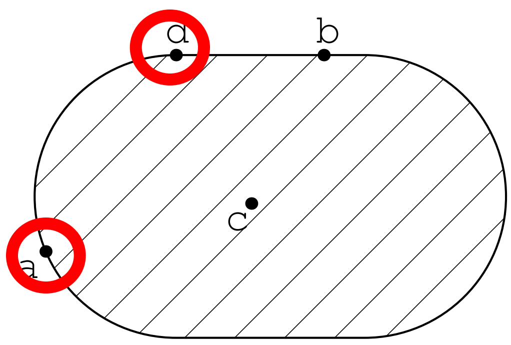
k-extreme: not interior to k+1 simplex in the body

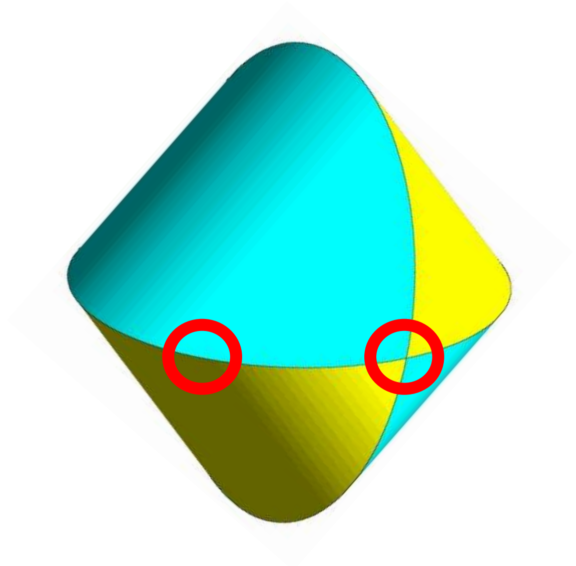




Convex extremity

k-extreme: not interior to k+1 simplex in the body

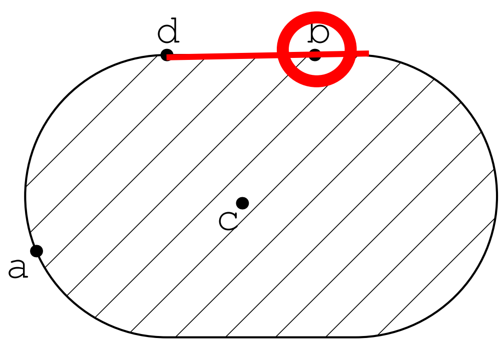


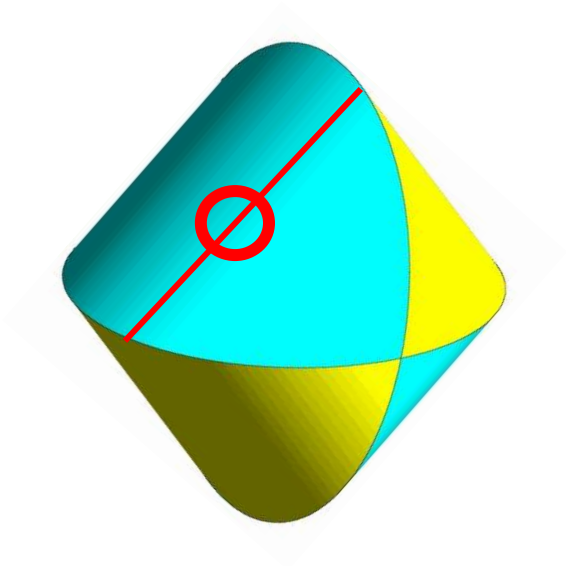


0-extreme

Convex extremity

k-extreme: not interior to k+1 simplex in the body

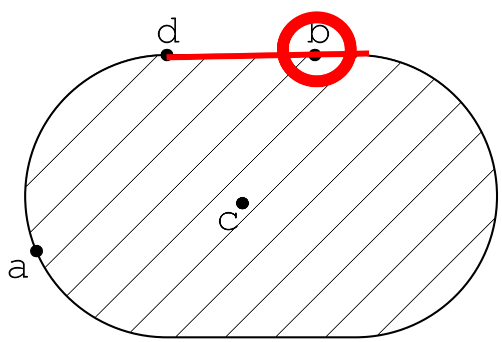


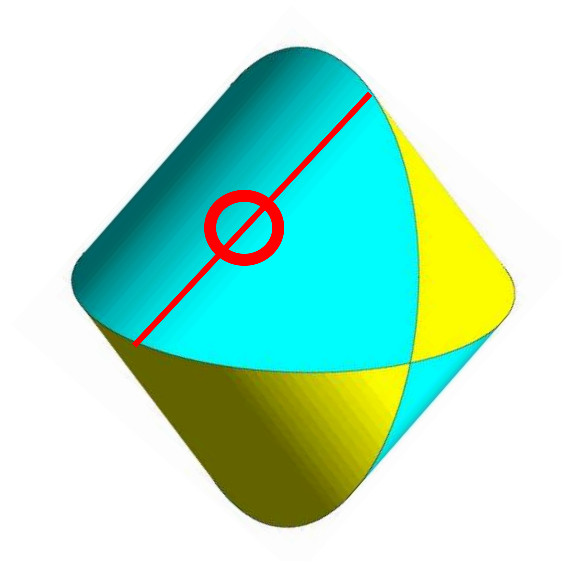


1-extreme

Convex extremity

k-extreme: not interior to k+1 simplex in the body
Face(x): largest convex subset with x in interior
(Dim Face (x) = smallest k st x is k-extreme)





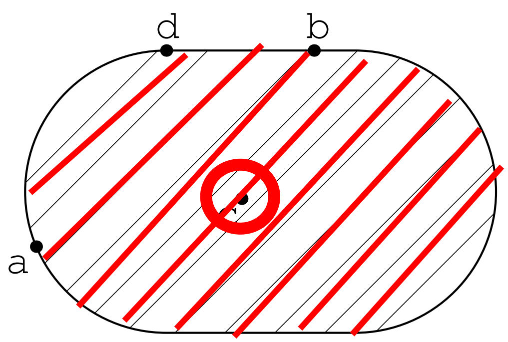
1-extreme

Convex extremity

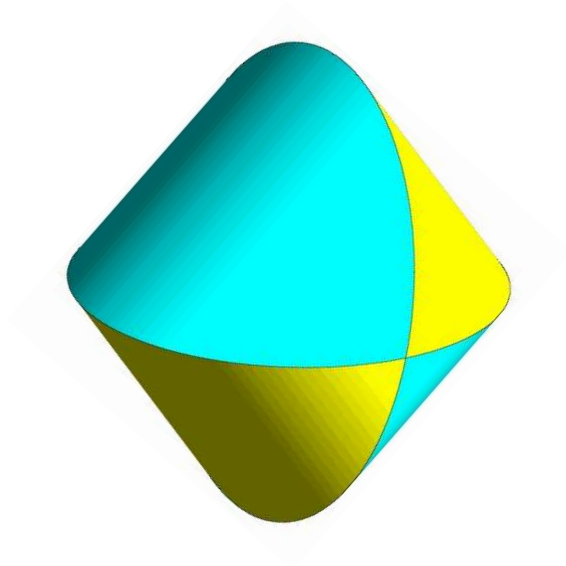
k-extreme: not interior to k+1 simplex in the body

Face(x): largest convex subset with x in interior

(Dim Face (x) = smallest k st x is k-extreme)

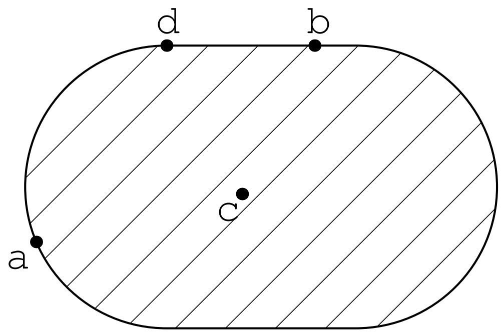


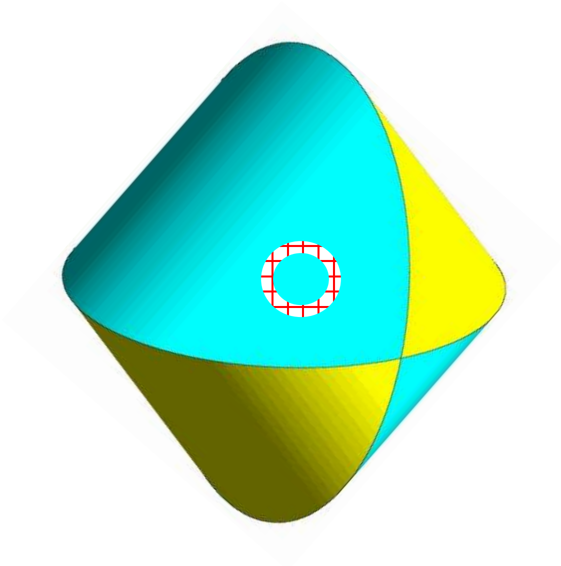
2-extreme



Convex extremity

k-extreme: not interior to k+1 simplex in the body
Face(x): largest convex subset with x in interior
(Dim Face (x) = smallest k st x is k-extreme)

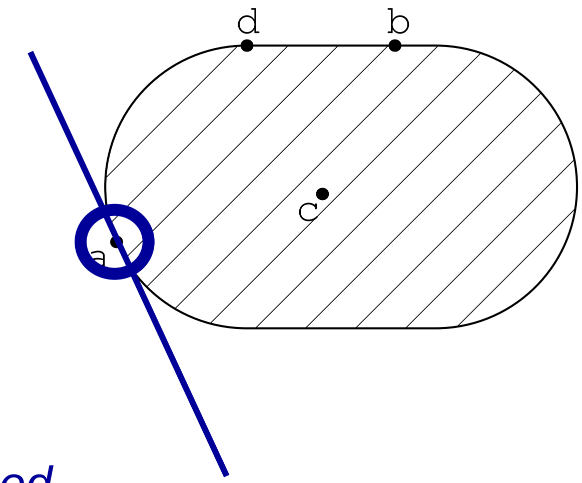




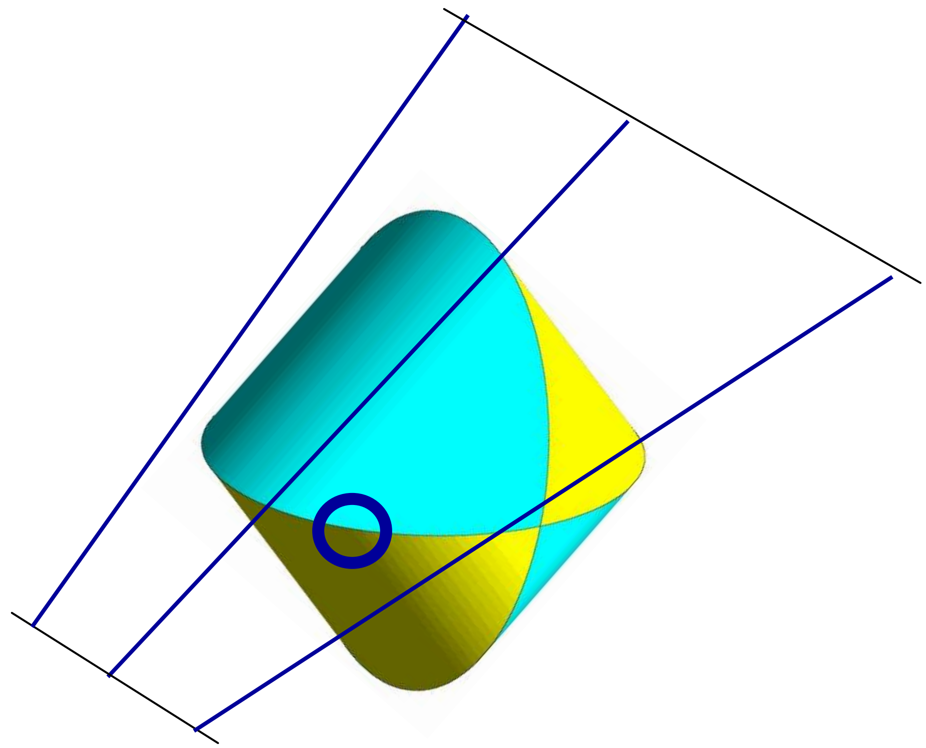
3-extreme

Convex exposedness

- k-exposed: exists halfspace with k dim intersection

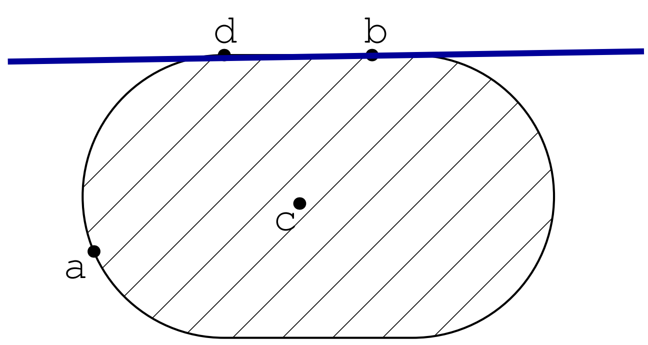


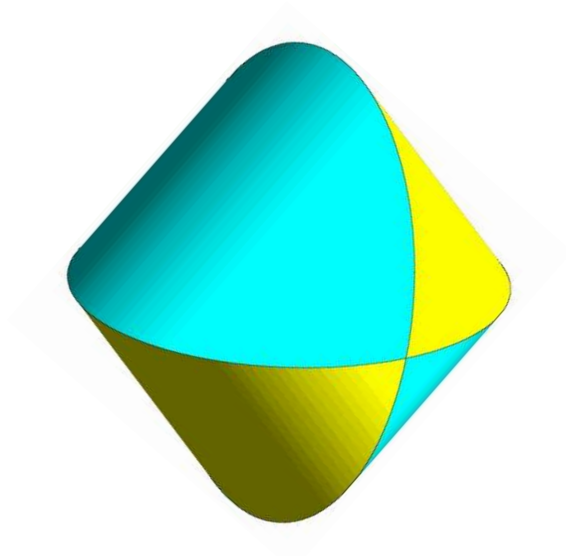
0-exposed



Convex exposedness

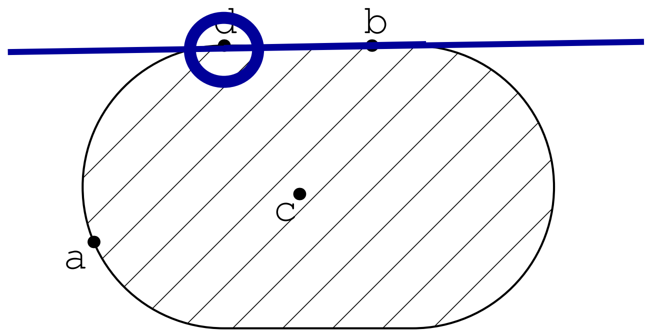
- Point d is 0-extreme (with 0-dim face)
- But not 0-exposed!
  - Every supporting hyperplane includes more stuff

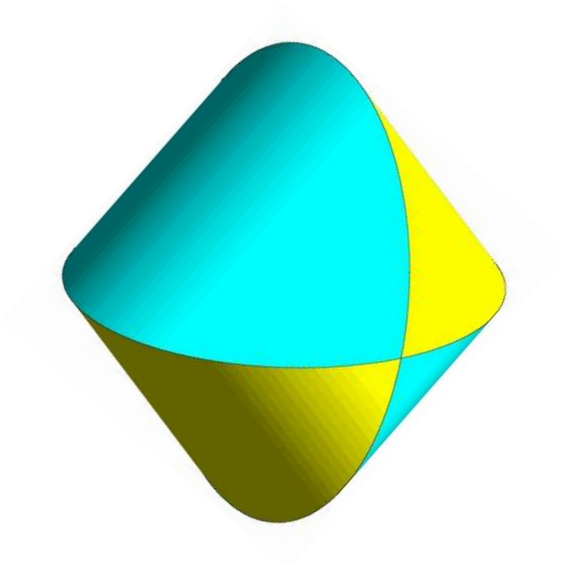




Convex exposedness

- k-exposed: exists halfspace with k dim intersection

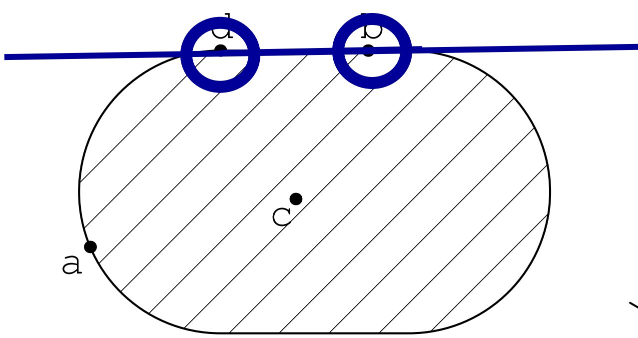


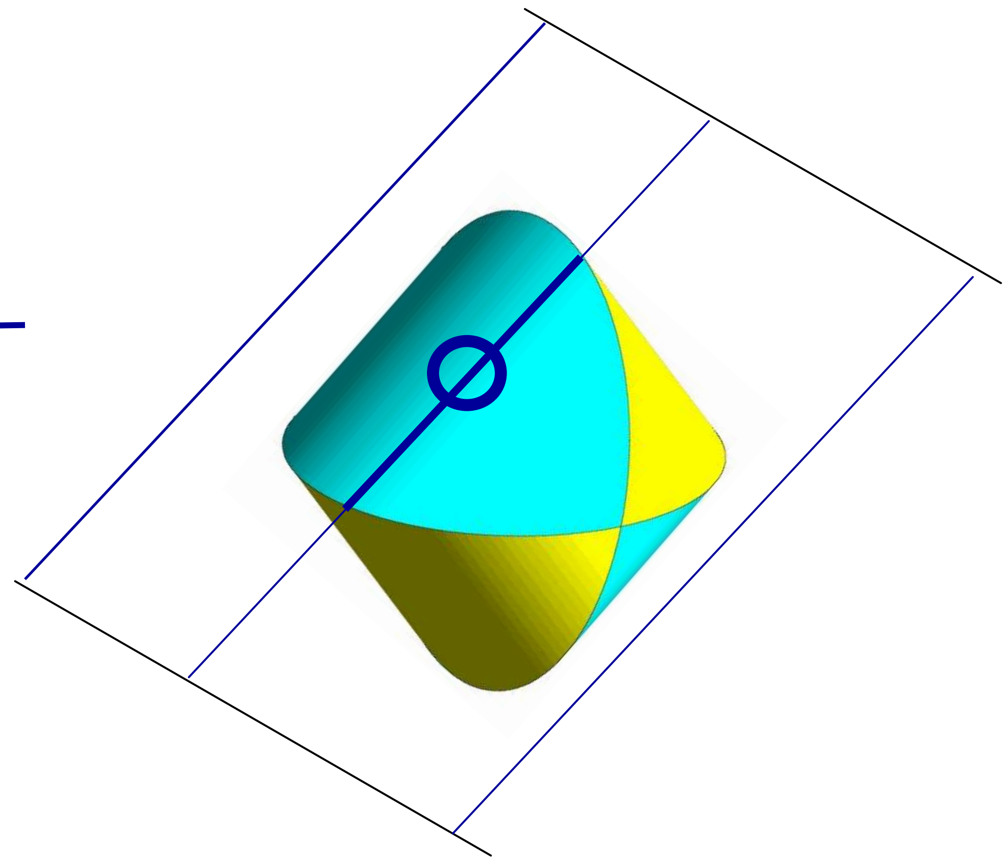


1-exposed

Convex exposedness

- k-exposed: exists halfspace with k dim intersection

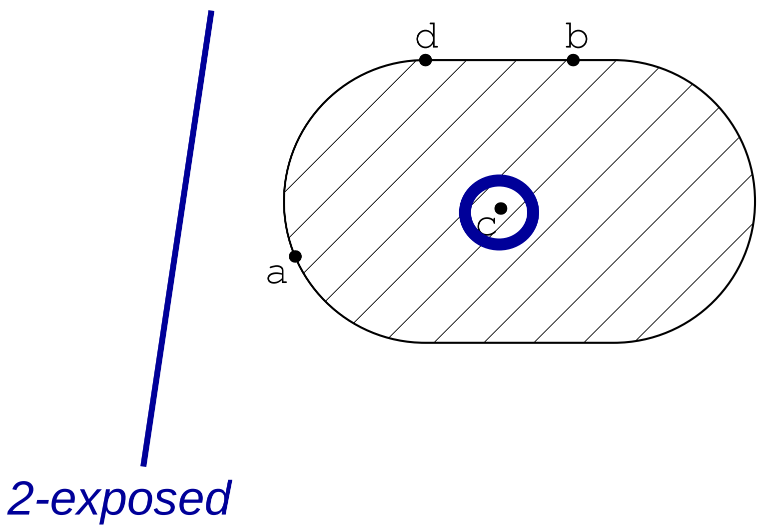


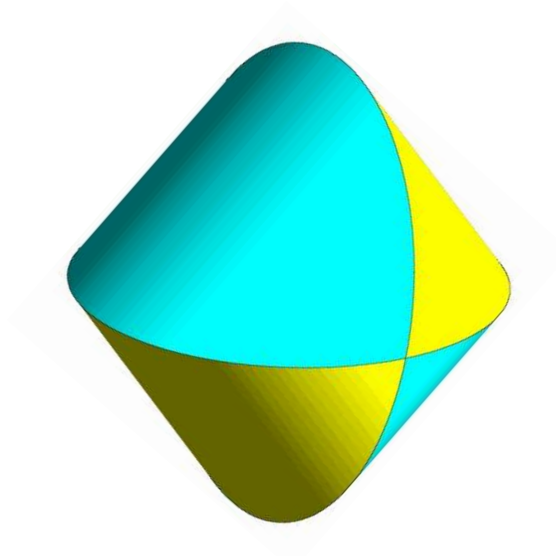


1-exposed

Convex exposedness

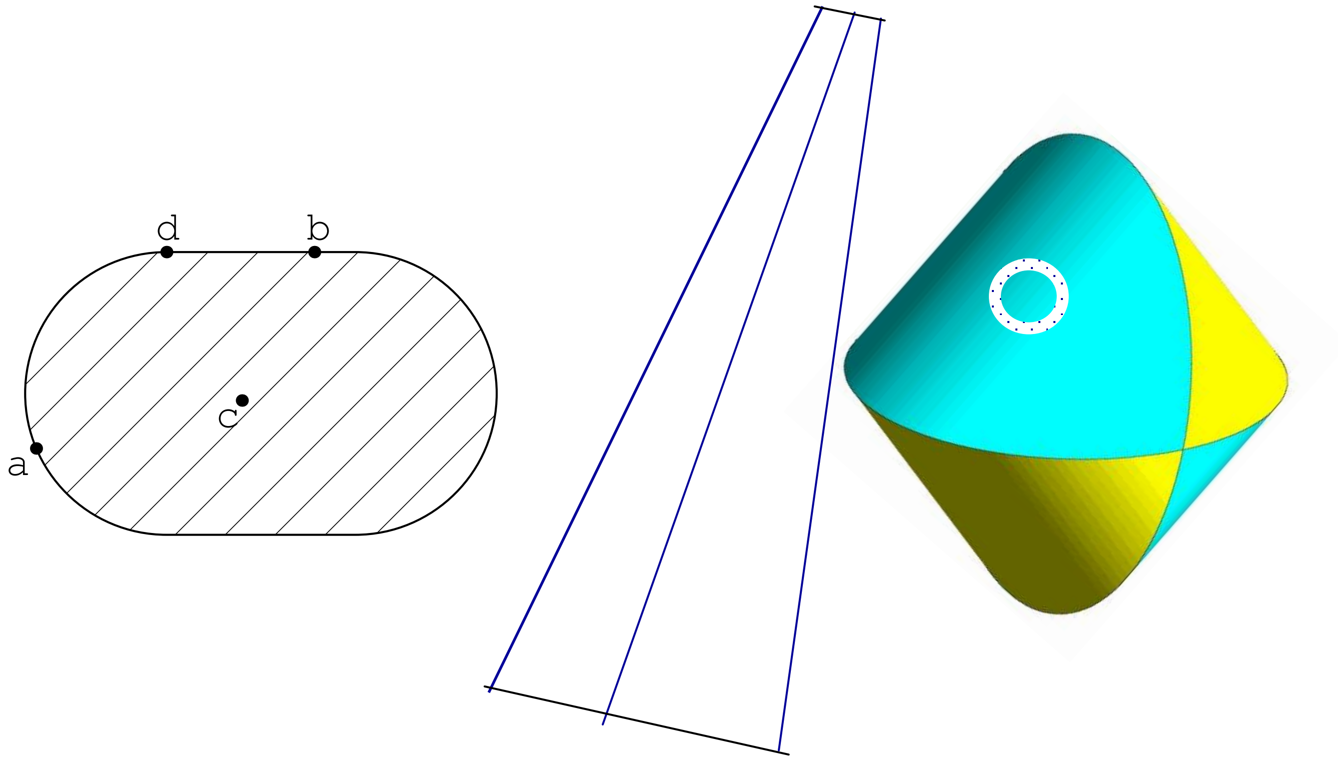
- k-exposed: exists halfspace with k dim intersection





Convex exposedness

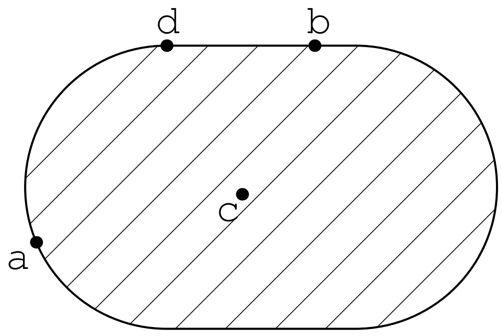
- k-exposed: exists halfspace with k dim intersection

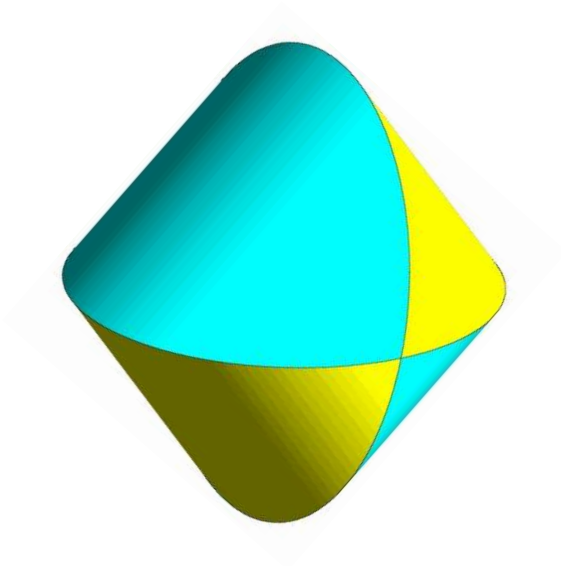


3-exposed

Convex exposedness

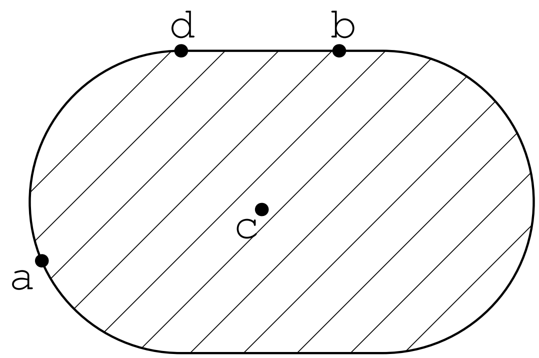
- k-exposed implies k-extreme
- But not vice-versa

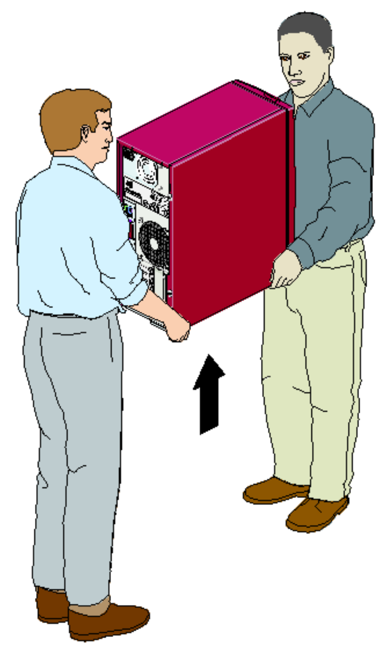




Straszewicz and Asplund

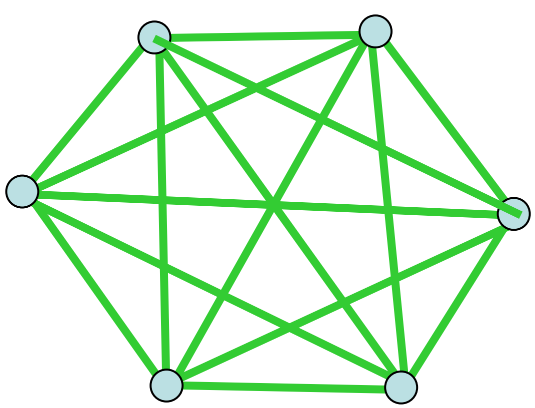
- [Stra 35]: The 0-exposed are dense in the 0-extreme
- [Asp 63]: The k-exposed are dense in the k-extreme





Onto our convex body

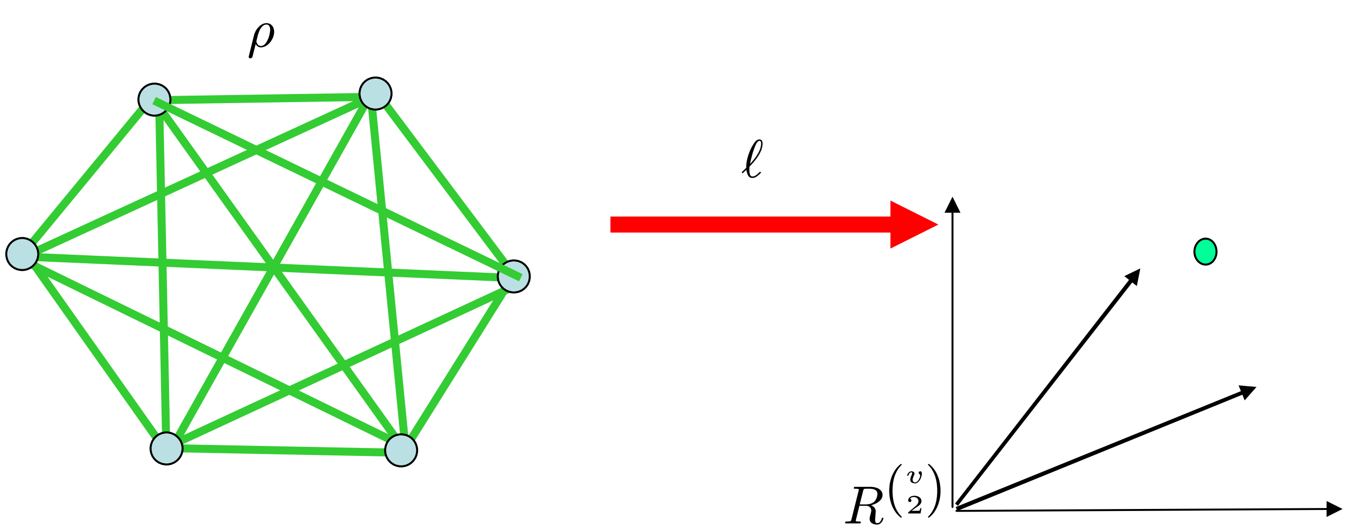
- We will be interested in our graph G
- But we will start with the complete graph on v-verts



Measuring the simplex

Δ_v: the simplex on v vertices

$\ell(\rho) \in R^{\binom{v}{2}}$: squared edge lengths

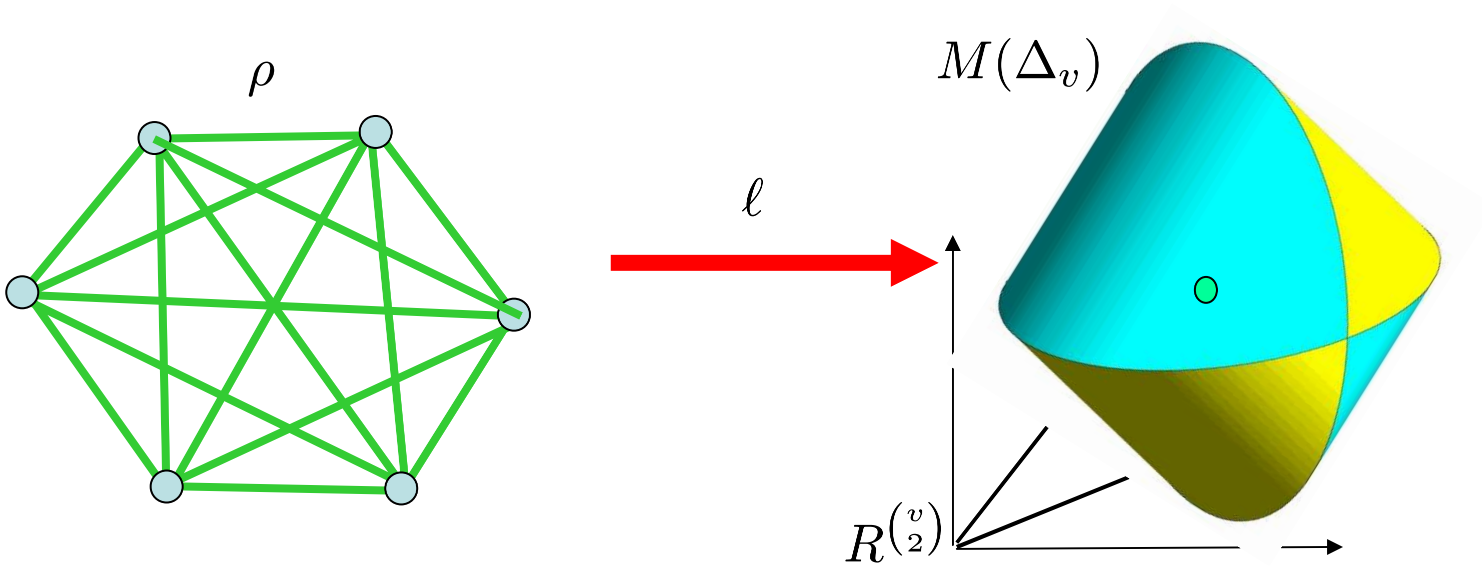


Measuring the simplex

Δ_v: the simplex on v vertices

$\ell(\rho) \in R^{\binom{v}{2}}$: squared edge lengths

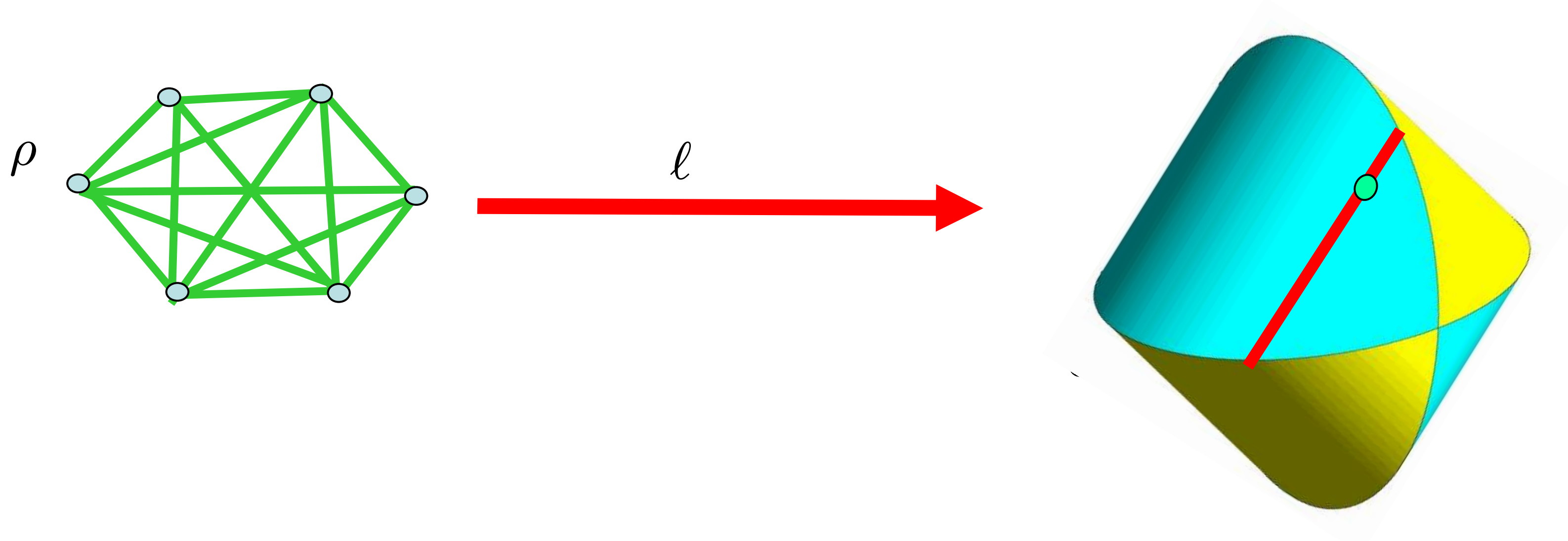
$M(\Delta_v)$: image of ℓ (any dim span)



Face structure of $M(\Delta_v)$

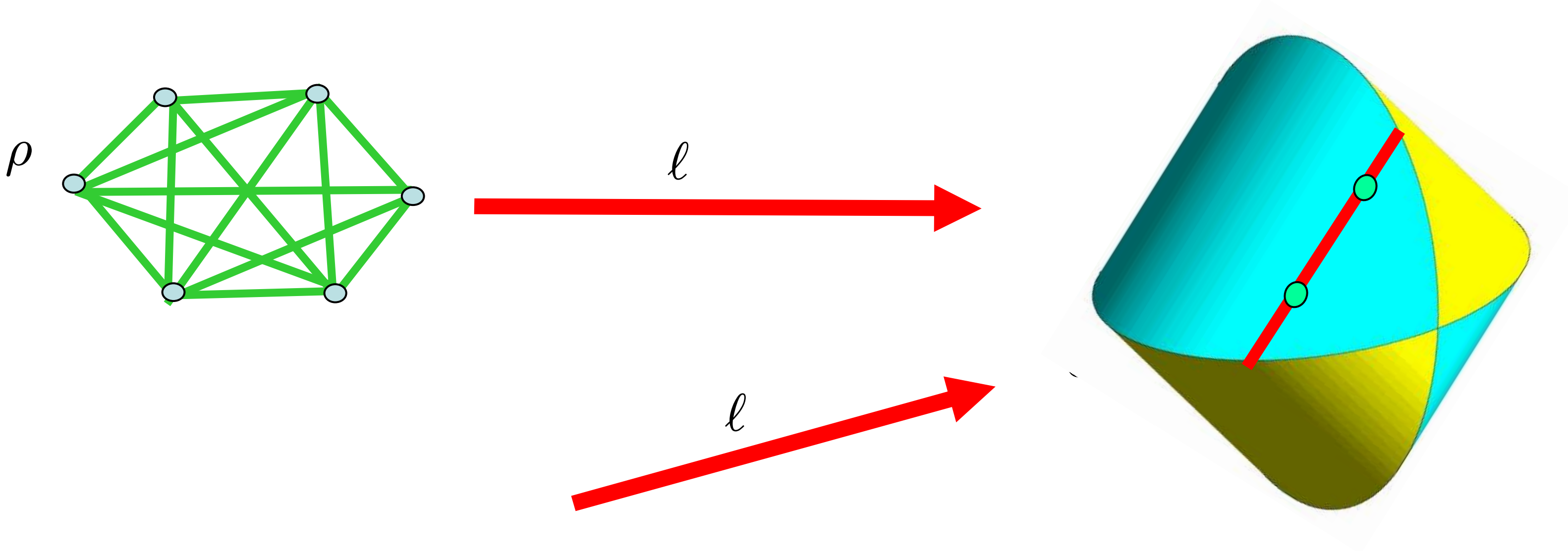
$M(\Delta_v)$ is isometric to cone of PSD (v-1) by (v-1) matrices

$l(q) \in Face(l(p)) ==$ q is affine image of p



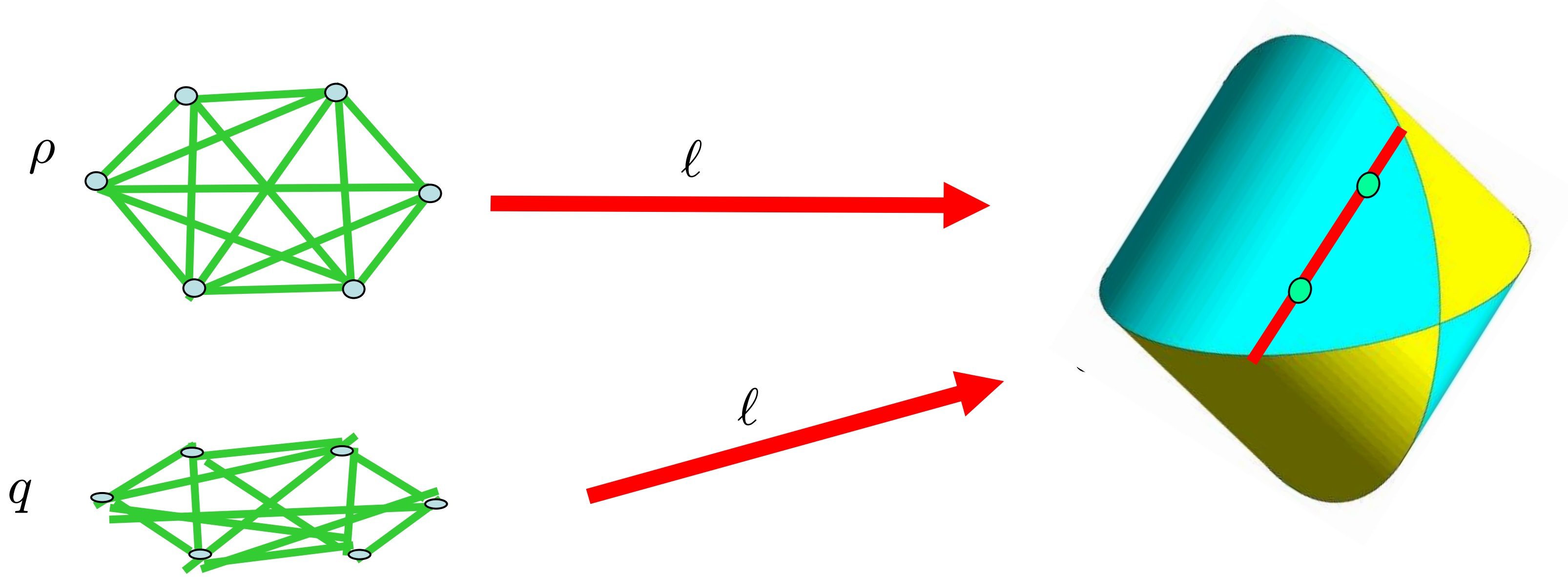
Face structure of $M(\Delta_v)$

$M(\Delta_v)$ is isometric to cone of PSD (v-1) by (v-1) matrices
$l(q) \in Face(l(p)) ==$ q is affine image of p



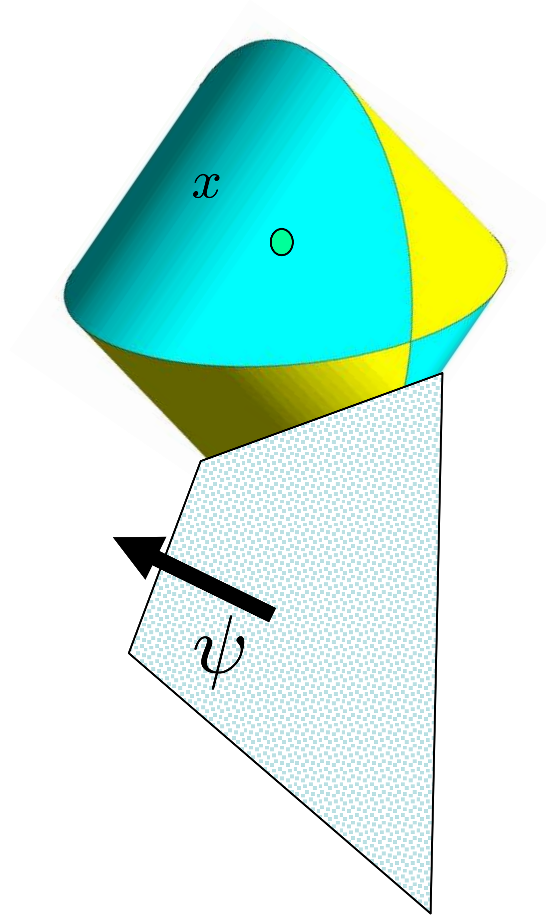
Face structure of $M(\Delta_v)$

$M(\Delta_v)$ is isometric to cone of PSD (v-1) by (v-1) matrices
$l(q) \in Face(l(p)) ==$ q is affine image of p



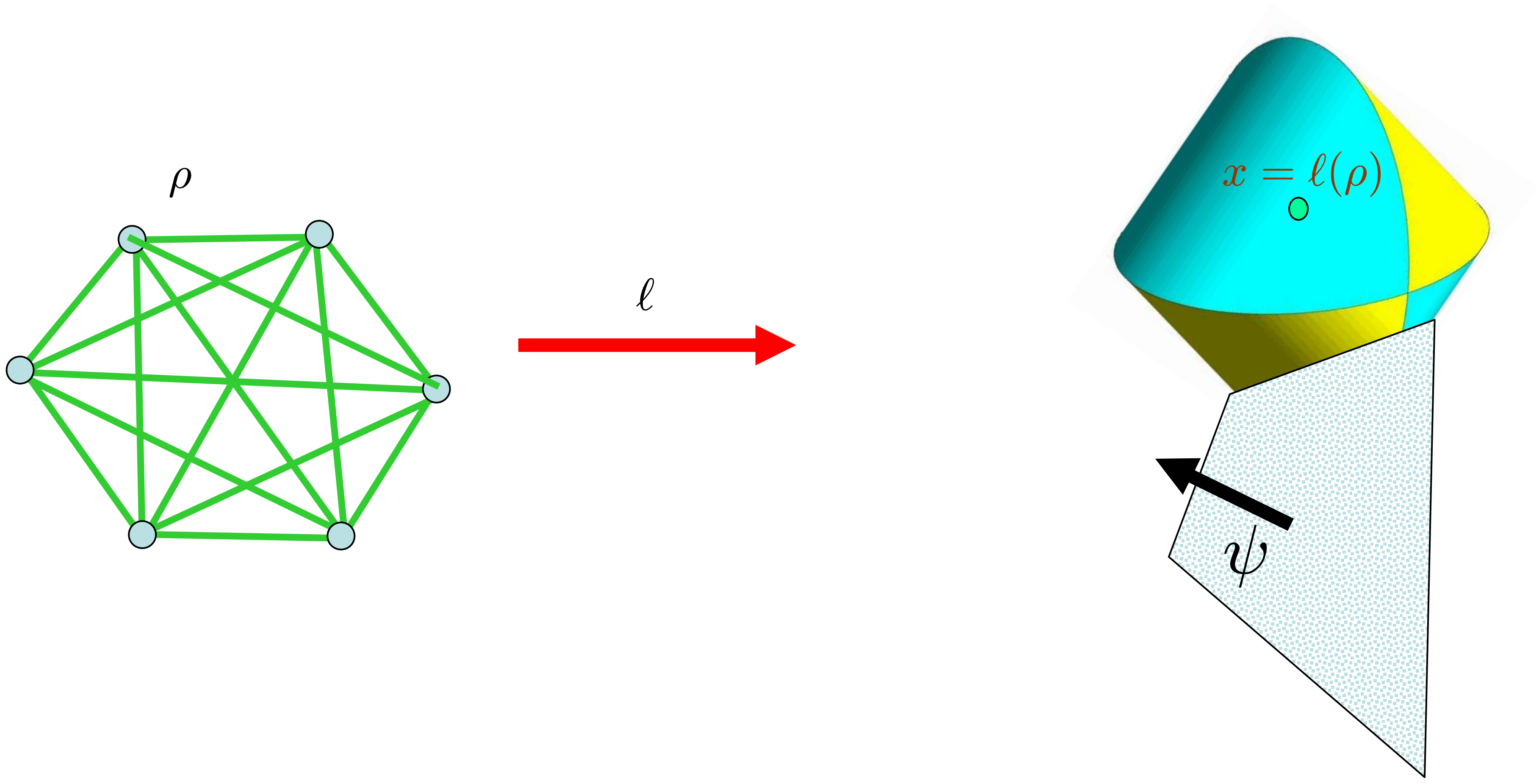
Hyperplane distance

$x \cdot \psi$



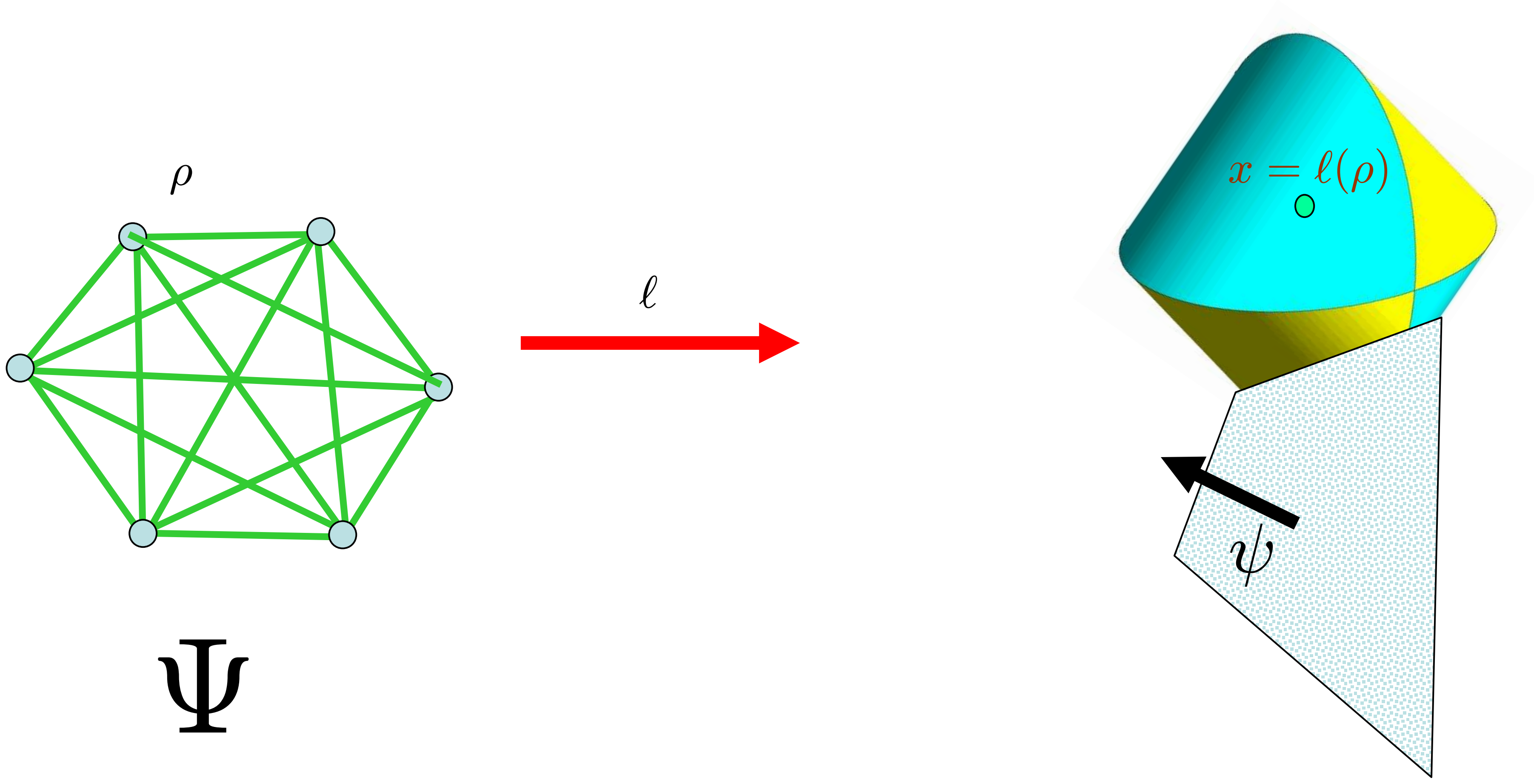
Pull Back

$$x \cdot \psi = \ell(\rho) \cdot \psi$$



Matricize

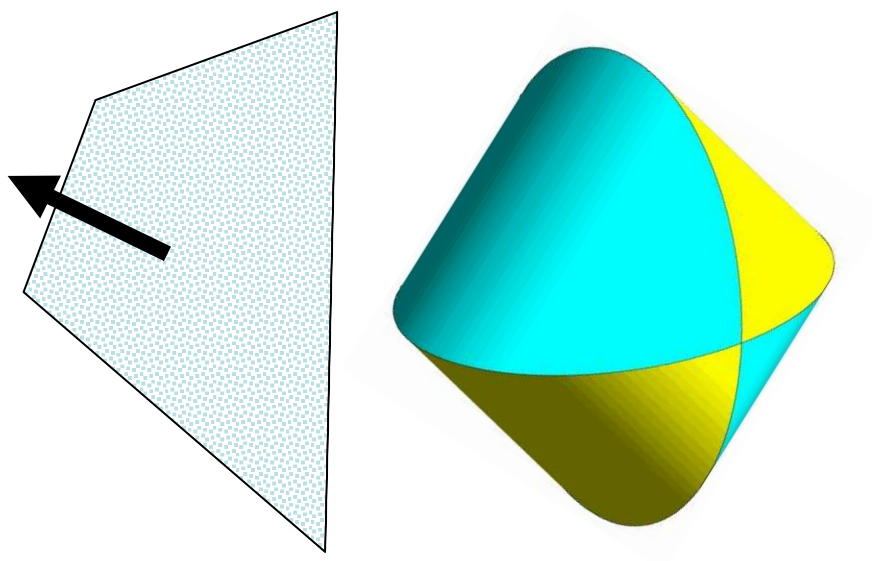
$$x \cdot \psi = \ell(\rho) \cdot \psi = \sum_j (\rho^j)^t \Psi \rho^j$$



Bounding Hyperplane

$\forall \rho \quad 0 \leq \sum_j (\rho^j)^t \Psi \rho^j$

Ψ is PSD



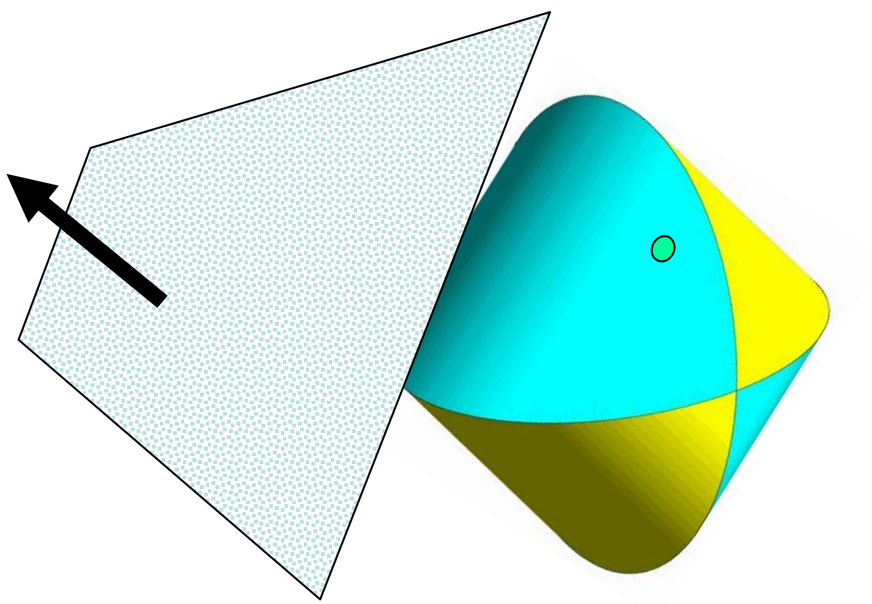
Supporting Hyperplane

Ψ is PSD. $\sum_j (\rho^j)^t \Psi \rho^j = 0$

$\Psi \rho^j = 0$

ρ^j is in the kernel

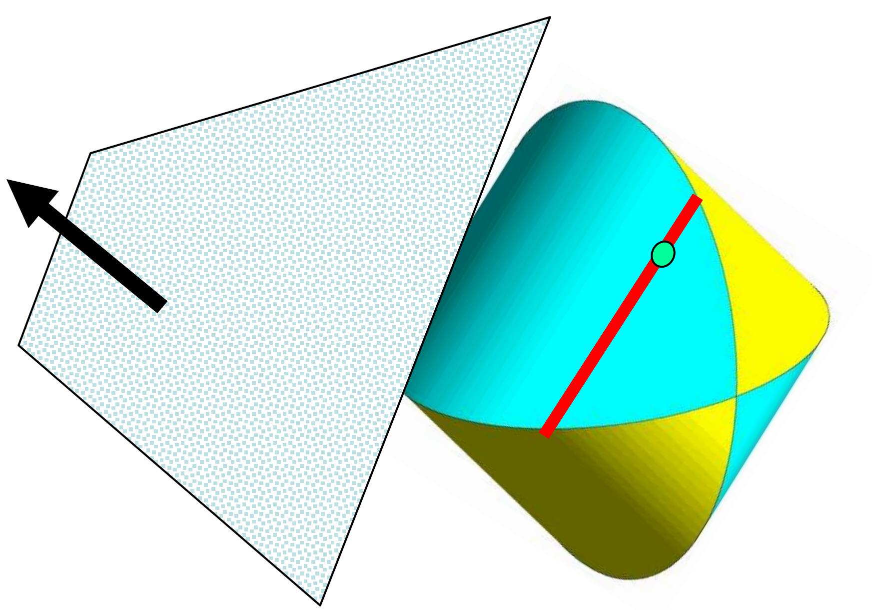
PSD contact == kernel == stress



Supporting Hyperplane

PSD contact at ρ

Contact at the face of $\ell(\rho)$ (at least)

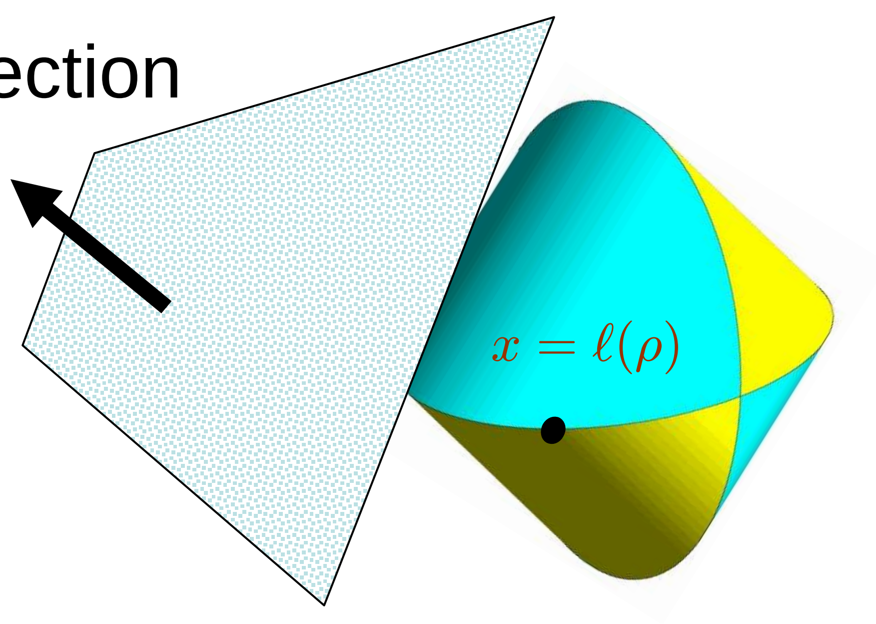


Starting the argument

- We assume that p is generic framework of G and is UR
- We want to find a PSD ESM with rank v-d-1

What do we need

- PSD ESM matrix
  - Need a supporting hyperplane of $M(\Delta_v)$
  - With intersection at x
- Rank v-d-1
  - Only affines of p are in kernel
  - Hyperplane intersection exactly $\mathrm{Face}\ \ell(\rho)$



What else do we need

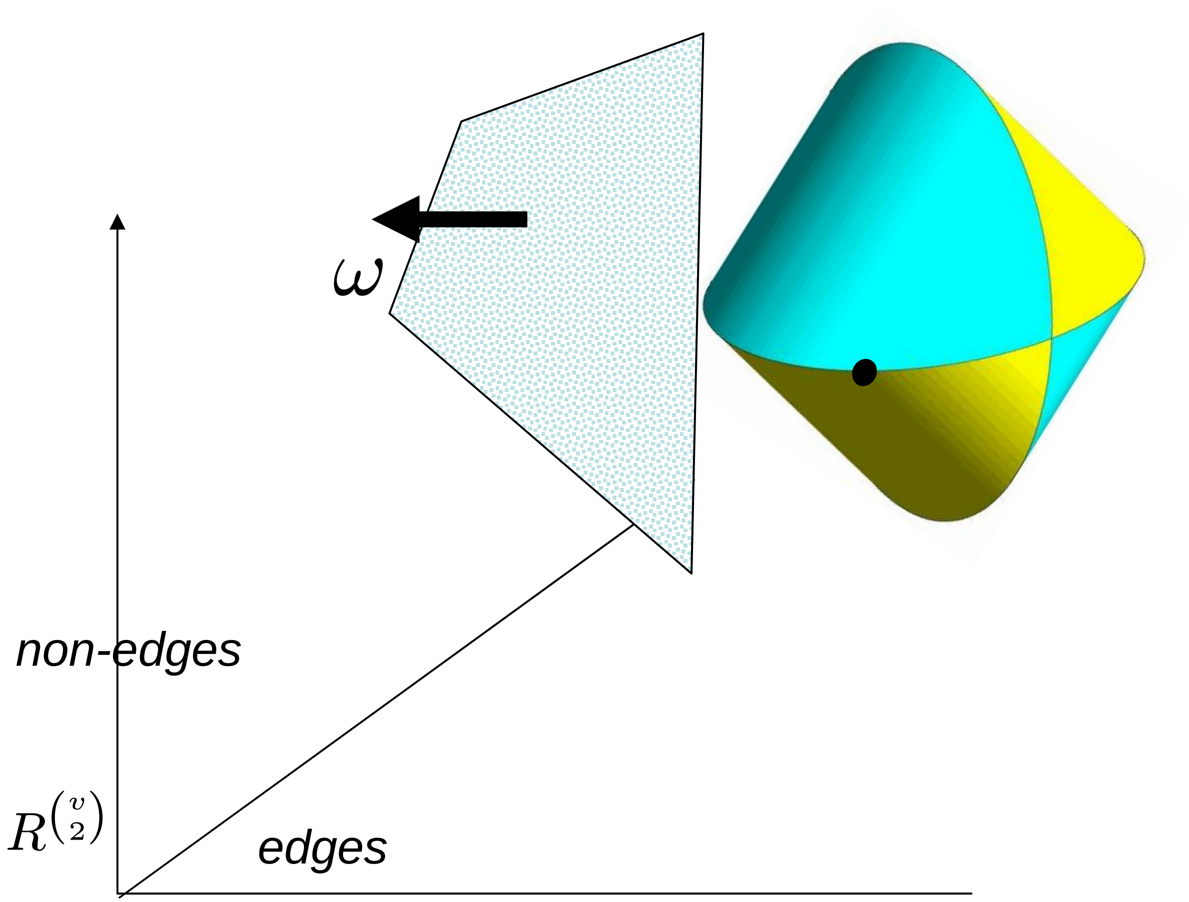
- Recall that we are interested in the graph G, not the simplex

What else do we need

- We need zeros at non edges of G

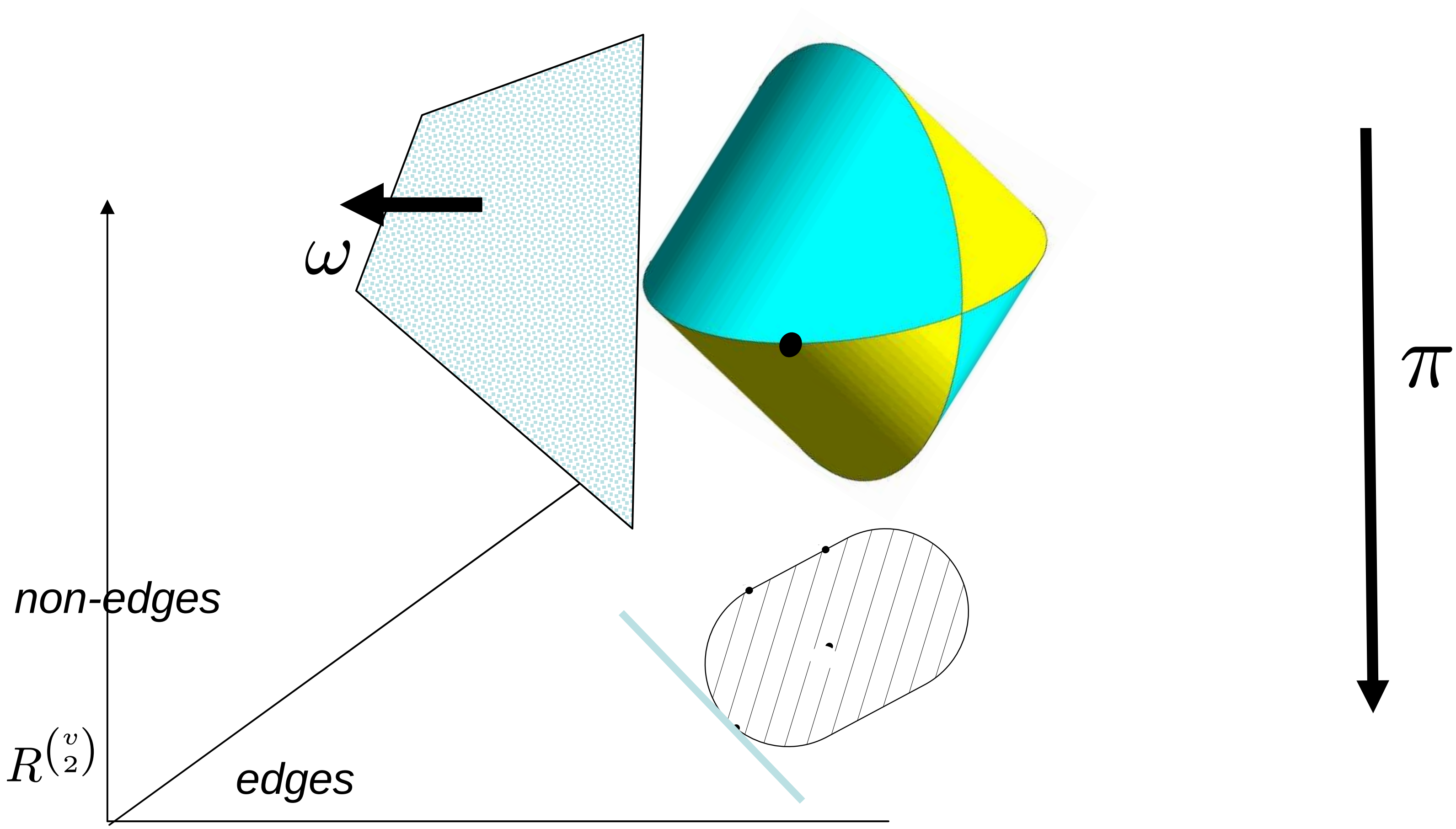
Lets model this

- We need zeros at non edges of G



Lets model this

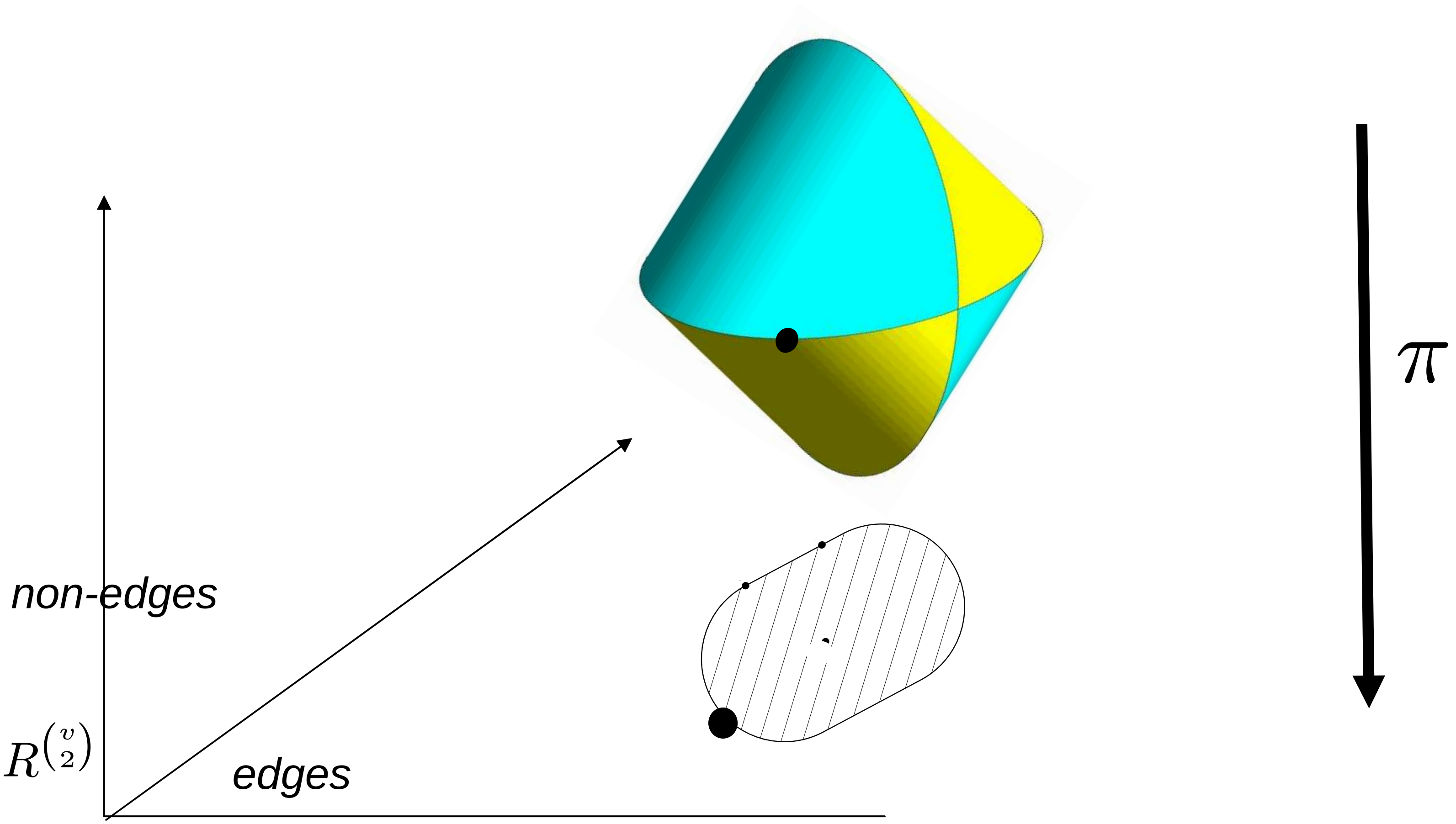
- We need zeros at non edges of G
  - Projects down to a hyperplane



Faces correspond

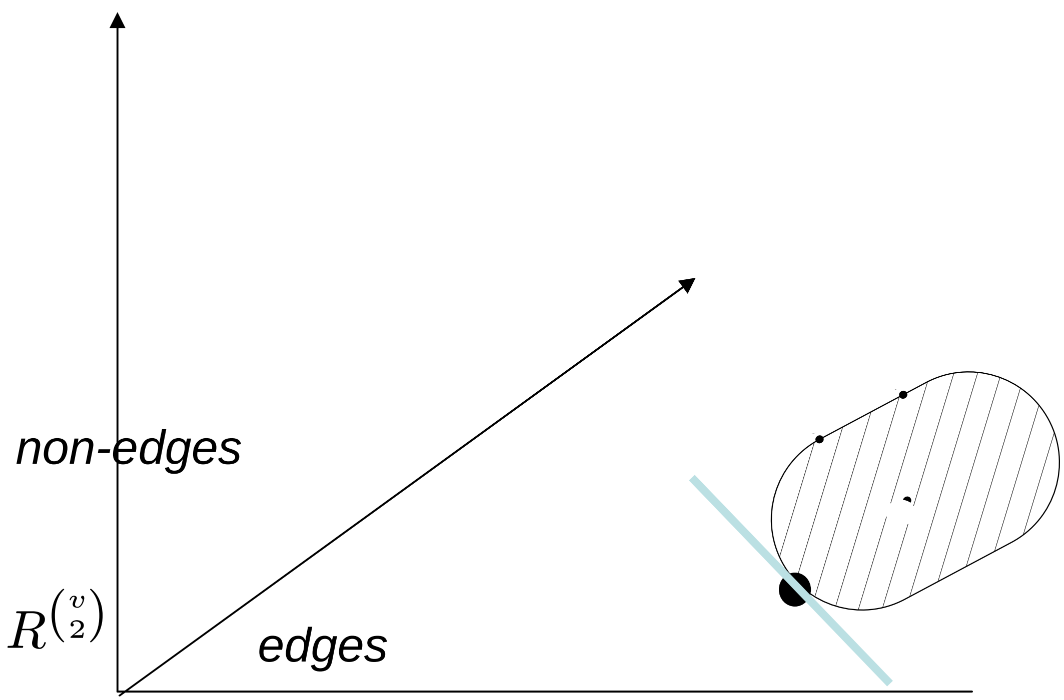
Lemma:

Universal rigidity: $\pi^{-1}(Face(\pi(x))) = Face(x)$



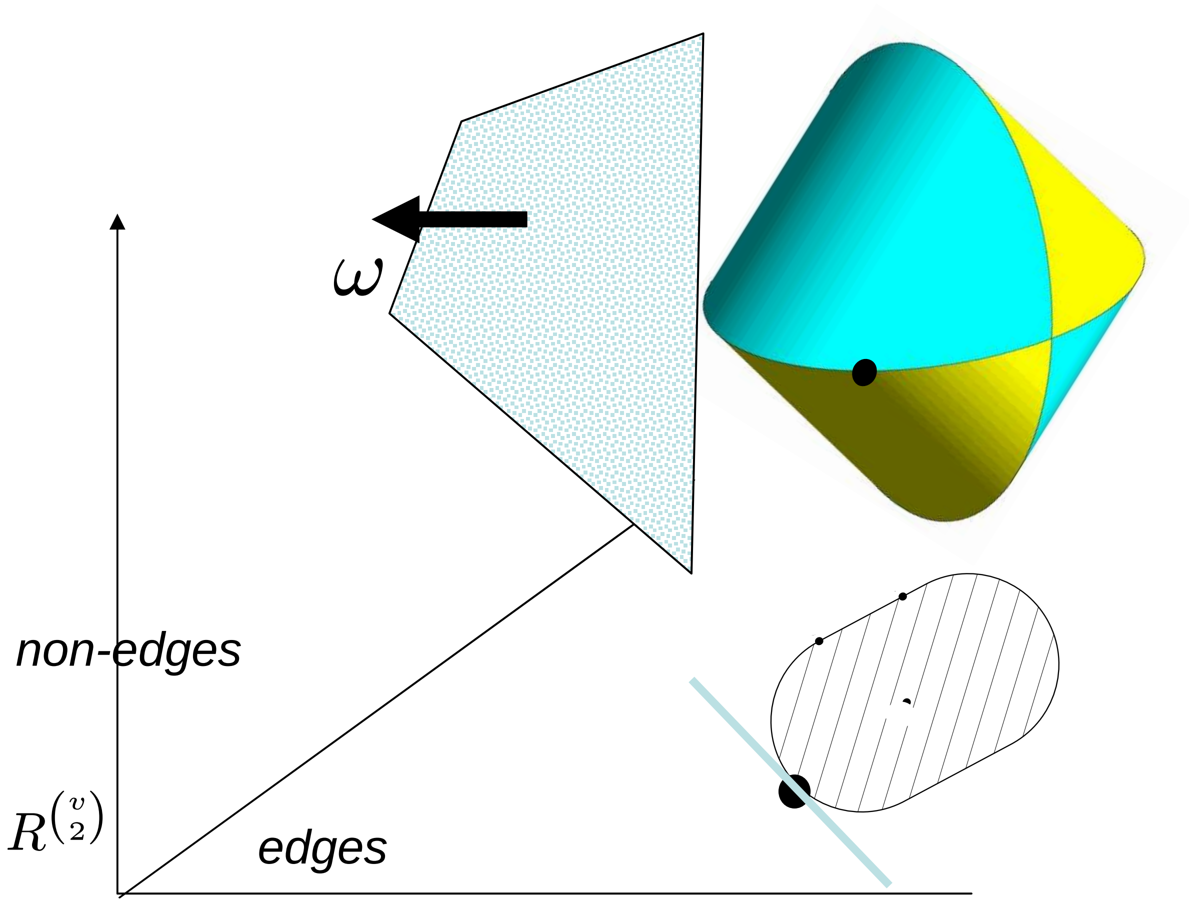
So we really just need

$$\text{Any hyperplane downstairs with support Face}(\pi(\ell(\rho)))$$



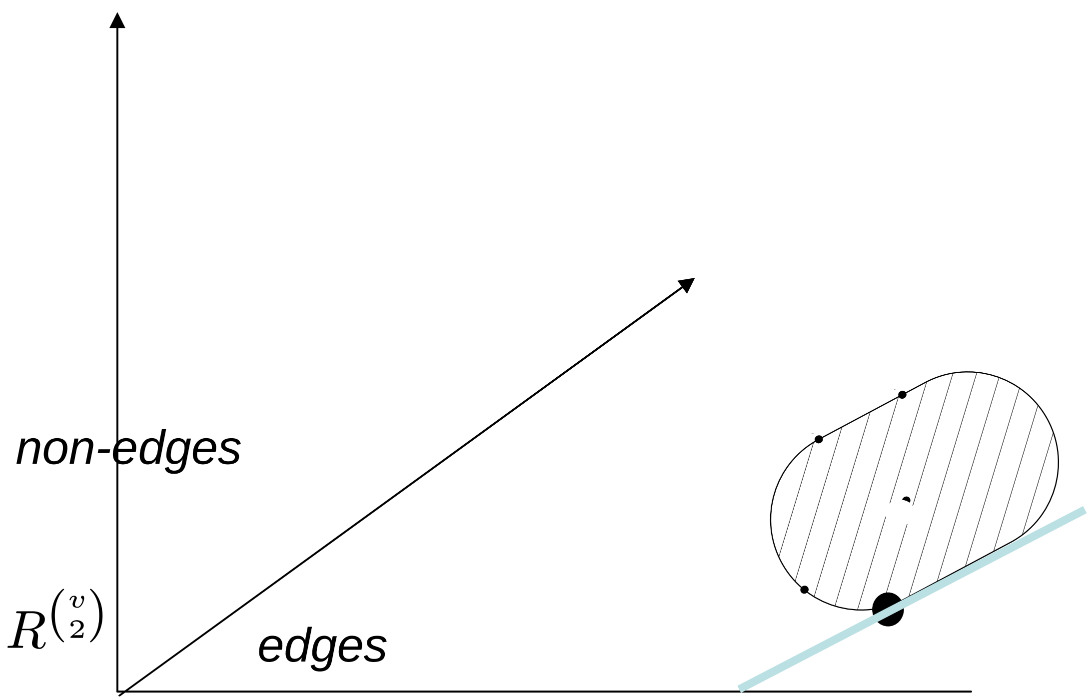
So we really just need

Any hyperplane downstairs with support $\mathrm{Face}(\pi(\ell(\rho)))$ upstairs has support $\mathrm{Face}(\ell(\rho))$



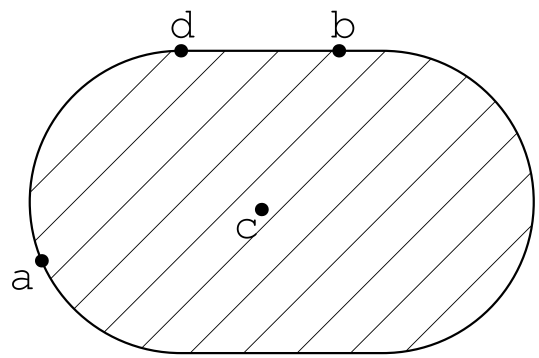
What Could go Wrong

Any hyperplane downstairs with support $\mathrm{Face}(\pi(\ell(\rho)))$



Straszewicz and Asplund

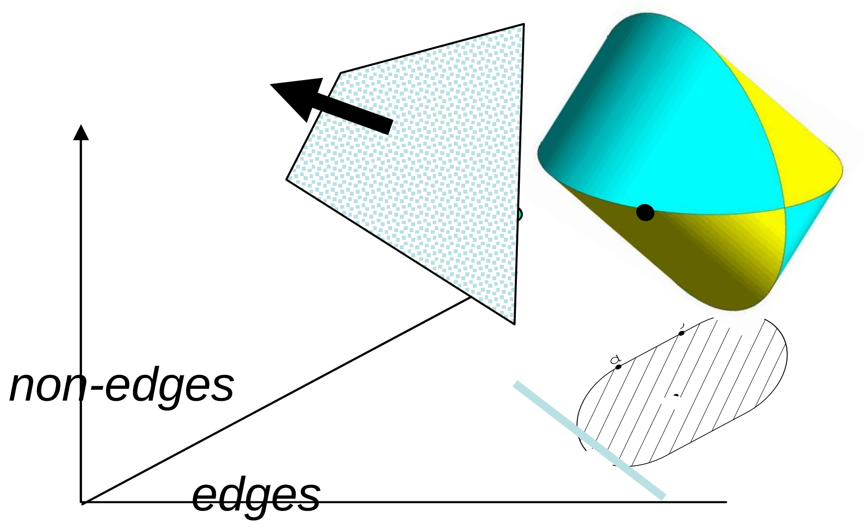
- [Stra 35]: The 0-exposed are dense in the 0-extreme
- [Asp 63]: The k-exposed are dense in the k-extreme



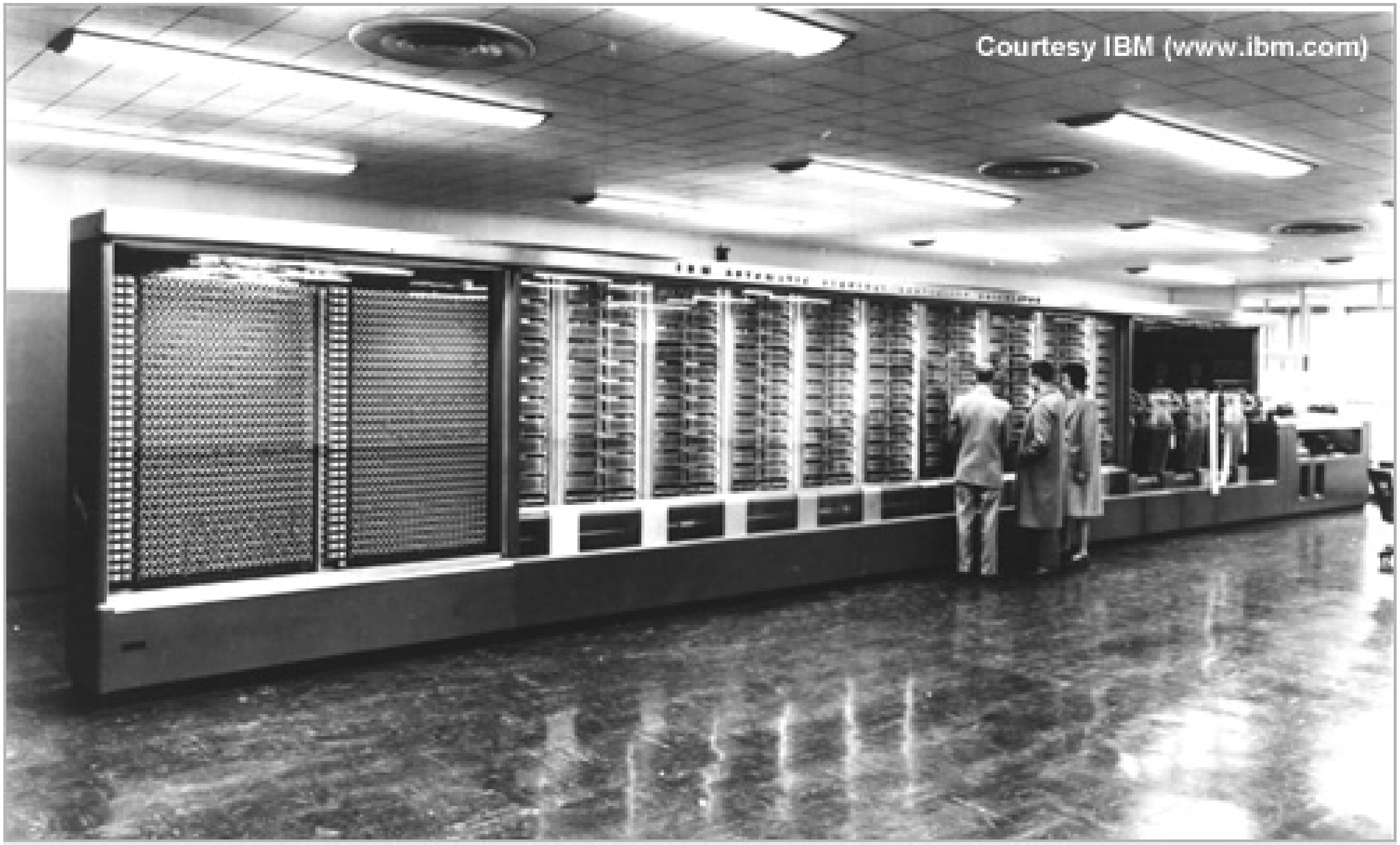
- Use genericity of p (+ a few technicalities)

Now we are done

- Generic point upstairs, maps to generic downstairs
- Downstairs point has supporting plane with right contact (Asplund)
- Corresponds to PSD stress of G
- Contact dim upstairs gives right rank



Computation and SDP



The Harvard Mark I

Gram UR testing

- Suppose generic d-dimensional framework
- Given distances, set up SDP feasibility to look for embeddings
  - Complexity of (exact) SDP feasibility unknown
- UR => only rank(x)==d+1

Lin span = aff span + 1

Gram UR testing

- Suppose generic d-dimensional framework
- Given distances, set up SDP feasibility to look for embeddings
  - Complexity of (exact) SDP feasibility unknown
- UR => only rank(x)=d+1
- ....details.....
- Not UR => there is x with rank > d+1
- UR: highest rank x is d+1

Stress UR testing

- Suppose generic d-dimensional framework
- Given distances, can set up SDP feasibility to look for PSD stresses
- UR: highest rank (Ω) is v-d-1 (thm)

Stress UR testing

- Suppose generic d-dimensional framework
- Given distances, can set up SDP feasibility to look for PSD stresses
- UR: highest rank (Ω) is v-d-1 (thm)
- So is it any easier??

Dual Computation

- Suppose generic d-dimensional framework
- Given distances, can set up SDP feasibility to look for PSD stresses
  - This is simply "the dual" SDP

Dual Computation

- Suppose generic d-dimensional framework
- Given distances, can set up SDP feasibility to look for PSD stresses
  - This is simply “the dual” SDP
  - UR => highest rank x is d+1
  - UR => highest rank Ω is v-d-1 (thm)

Dual Computation

- Suppose generic d-dimensional framework
- Given distances, can set up SDP feasibility to look for PSD stresses
  - This is simply “the dual” SDP
  - UR => highest rank x is d+1
  - UR => highest rank Ω is v-d-1 (thm)
  - UR => highest [rank(x)+rank(Ω)] is v (thm)
    - (x, Ω) satisfy strict complementarity if UR (thm)

Strict complementarity

$$\text{Min } \langle x, \beta \rangle$$
$$x \in L + b$$
$$x \in S^n_+$$

primal

$$\text{Min } \langle b, \Omega \rangle$$
$$\Omega \in L^\perp + \beta$$
$$\Omega \in S^n_+$$

dual

Strict complementarity

$$\begin{aligned}&\text{Min } \langle x, \beta \rangle\\ &x \in L + b\\ &x \in S^n_+\end{aligned}$$

primal

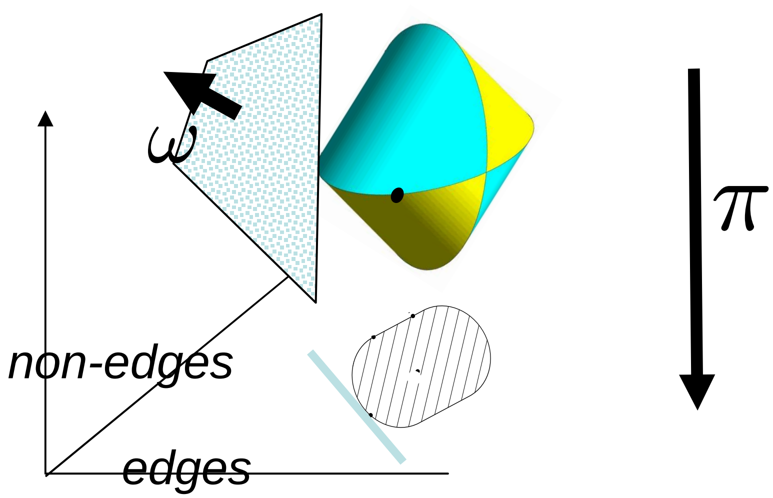
$$\begin{aligned}&\text{Min } \langle b, \Omega \rangle\\ &\Omega \in L^\perp + \beta\\ &\Omega \in S^n_+\end{aligned}$$

dual

- WC: $rank(x^*) + rank(\Omega^*) \leq n$
  - Holds for all solution pairs
- SC: $(x^*, \Omega^*) \mid rank(x^*) + rank(\Omega^*) = n$
  - Strong form of duality
  - Does not always hold for SDP
  - Needed for fast convergence

What we really have proven…

- We did not rely on any properties of the specific projection.



Strict complementarity

$$\text{Min } \langle x, \beta \rangle$$
$$x \in L + b$$
$$x \in S_+^n$$

$$\text{Min } \langle b, \Omega \rangle$$
$$\Omega \in L^\perp + \beta$$
$$\Omega \in S_+^n$$

[GT 09] $\forall (L, b, \beta) \mid x^*$ is unique and generic in its rank over $\mathbb{Q}(L, \beta)$,
$\exists \Omega^* \mid (x^*, \Omega^*)$ SC

UR

Generic d-dim framework

Compare to [AHO 97]

$$\text{Min } \langle x, \beta \rangle$$
$$x \in L + b$$
$$x \in S^n_+$$

$$\text{Min } \langle b, \Omega \rangle$$
$$\Omega \in L^\perp + \beta$$
$$\Omega \in S^n_+$$

[GT 09] $\forall (L, b, \beta) \mid x^*$ is unique and generic in its rank over $\mathbb{Q}(L, \beta)$,

$\exists \Omega^* \mid (x^*, \Omega^*)$ SC

[AHO 97] $\forall L$ over $\mathbb{Q}$, $\forall$ generic (b, β),

$\exists (x^*, \Omega^*)$ SC

Random lengths= not low rank

Remaining Questions

- Complexity of UR testing
- Is there a single natural unifying statement for strict complementarity
- Graphs that are always UR
- Graphs that are sometimes UR

The Harvard Mark I

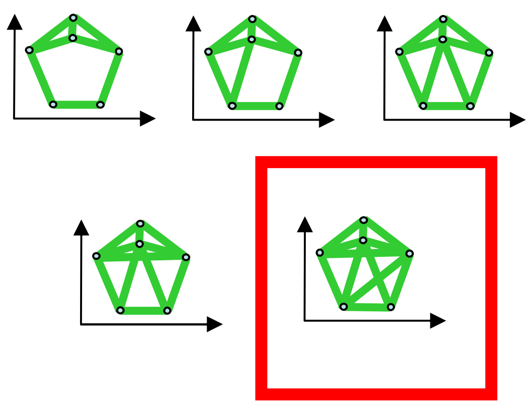
Remaining Questions

- Complexity of UR testing
- Is there a single natural unifying statement for strict complementarity
- Graphs that are always UR
- Graphs that are sometimes UR

[GT 09] $\forall (L, b, \beta) \mid x^*$ is unique and generic in its rank over $\mathbb{Q}(L, \beta)$,
$\exists \Omega^* \mid (x^*, \Omega^*)$ SC
[AHO 97] $\forall L$ over $\mathbb{Q}$, $\forall$ generic (b, β),
$\exists (x^*, \Omega^*)$ SC

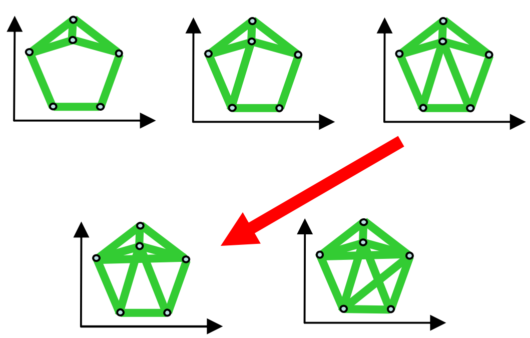
Remaining Questions

- Complexity of UR testing
- Is there a single natural unifying statement for strict complementarity
- Graphs that are always UR
- Graphs that are sometimes UR



Remaining Questions

- Complexity of UR testing
- Is there a single natural unifying statement for strict complementarity
- Graphs that are always UR
- Graphs that are sometimes UR



Thank You