THE PSD

MATRIX

A REVOLUTION ON MAX-CUT

The World as we know it

Linear Programming

$$\min_{x} \quad \mathbf{c}^\top \mathbf{x}$$
$$s.t. \quad A\mathbf{x} \leq \mathbf{b}, \ \mathbf{x} \geq 0$$

Integer Programming

$$\min_{x} \quad \mathbf{c}^\top \mathbf{x}$$
$$s.t. \quad A\mathbf{x} \leq \mathbf{b}, \ \mathbf{x} \in \mathbb{Z}$$

Quadratic Programming

$$\min_{x} \quad \frac{1}{2}\mathbf{x}^\top Q\mathbf{x} + \mathbf{c}^\top \mathbf{x}$$
$$s.t. \quad A\mathbf{x} \leq \mathbf{b}$$

Semidefinite Programming

The World as we know it

Linear Programming

$$\min_{x} \quad \mathbf{c}^\top \mathbf{x}$$
$$s.t. \quad A\mathbf{x} \leq \mathbf{b}, \ \mathbf{x} \geq 0$$

Integer Programming

$$\min_{x} \quad \mathbf{c}^\top \mathbf{x}$$
$$s.t. \quad A\mathbf{x} \leq \mathbf{b}, \ \mathbf{x} \in \mathbb{Z}$$

Wake up Ne

Quadratic Programming

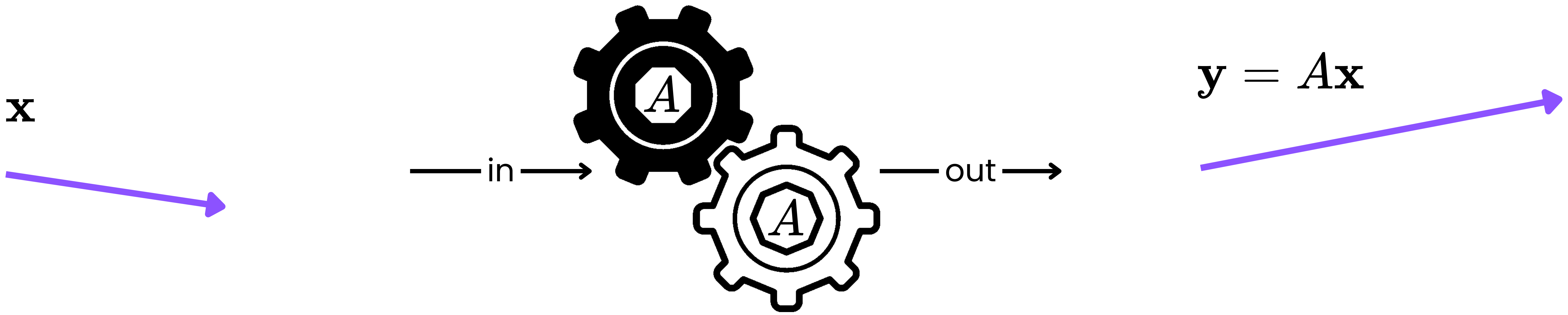
$$\min_{x} \quad \frac{1}{2}\mathbf{x}^\top Q\mathbf{x} + \mathbf{c}^\top \mathbf{x}$$
$$s.t. \quad A\mathbf{x} \leq \mathbf{b}$$

Semidefinite Programming

TIME TO TAKE THE PILL

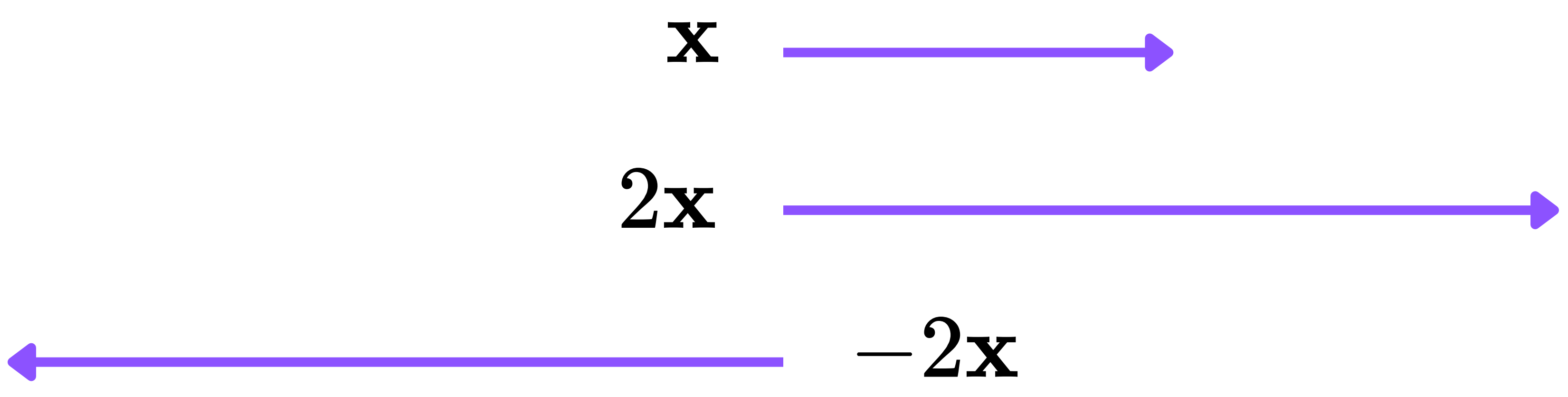
What’s The PSD Matrix?

What's a Matrix?



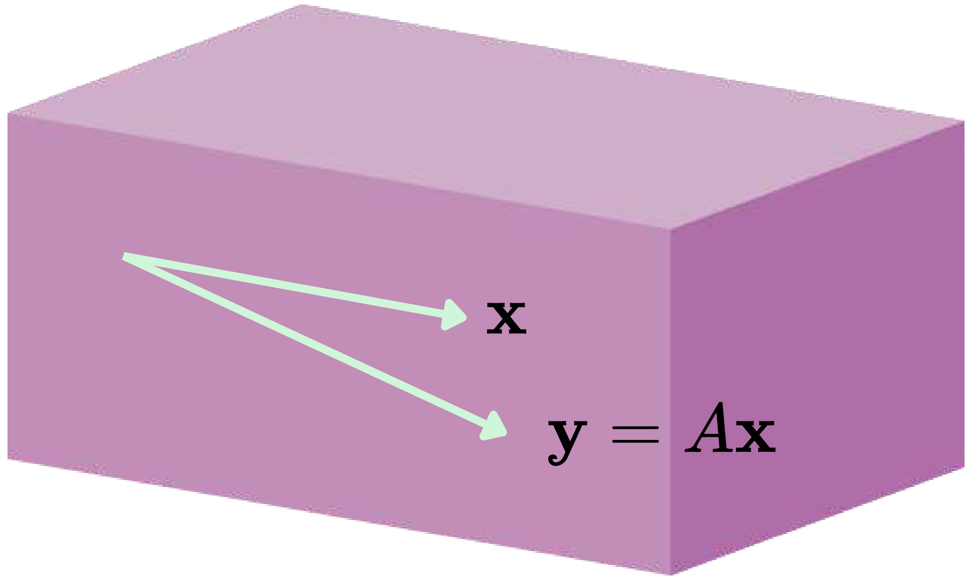
A **matrix** A is a **linear operator**

What does “positive” mean?



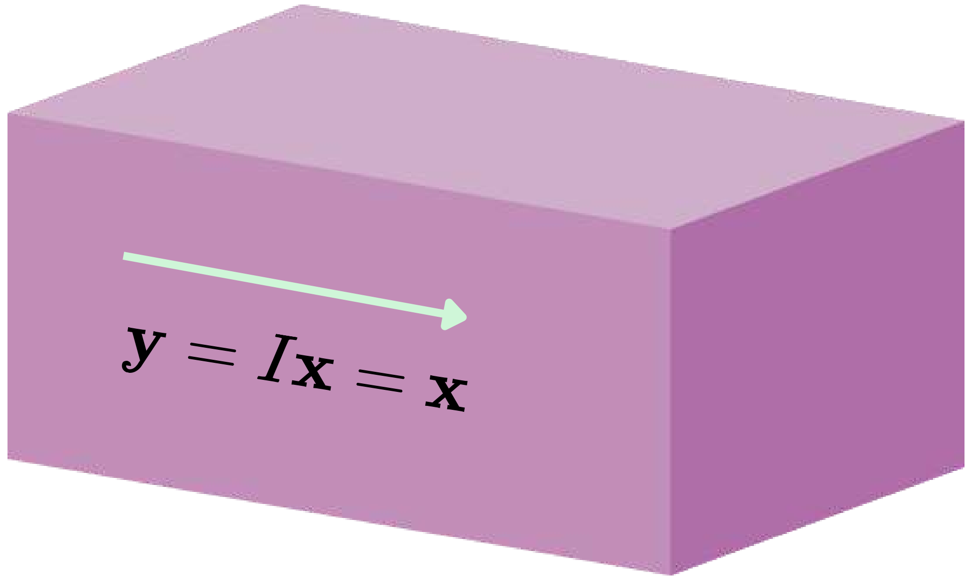
positive = on the same side

Positive Semi-Definite Matrix



Positive Semi-Definite:

$$A \geq 0 \quad \text{iff} \quad \forall \mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\},\ \mathbf{x}^\top \mathbf{y} = \mathbf{x}^\top A \mathbf{x} \geq 0.$$



Example

$$I > 0 : \mathbf{x}^\top I \mathbf{x} = \mathbf{x}^\top \mathbf{x} = \sum_i x_i^2 > 0, \quad \forall \mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}$$

positive definite

Positive Semi-Definite Matrix

$$\mathbf{x}^\top \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} \mathbf{x} = 1 \times \mathbf{x}_1^2 + 2 \times \mathbf{x}_2^2 > 0$$

Positive Definite

$$\mathbf{x}^\top \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \mathbf{x} = 1 \times \mathbf{x}_1^2 + 0 \times \mathbf{x}_2^2 \geq 0$$

Positive Semi-Definite

$$\mathbf{x}^\top \begin{bmatrix} 1 & 0 \\ 0 & -4 \end{bmatrix} \mathbf{x} = 1 \times \mathbf{x}_1^2 + -4 \times \mathbf{x}_2^2 \ngeq 0$$

Undefined

Is Positivity related to the Diagonal?

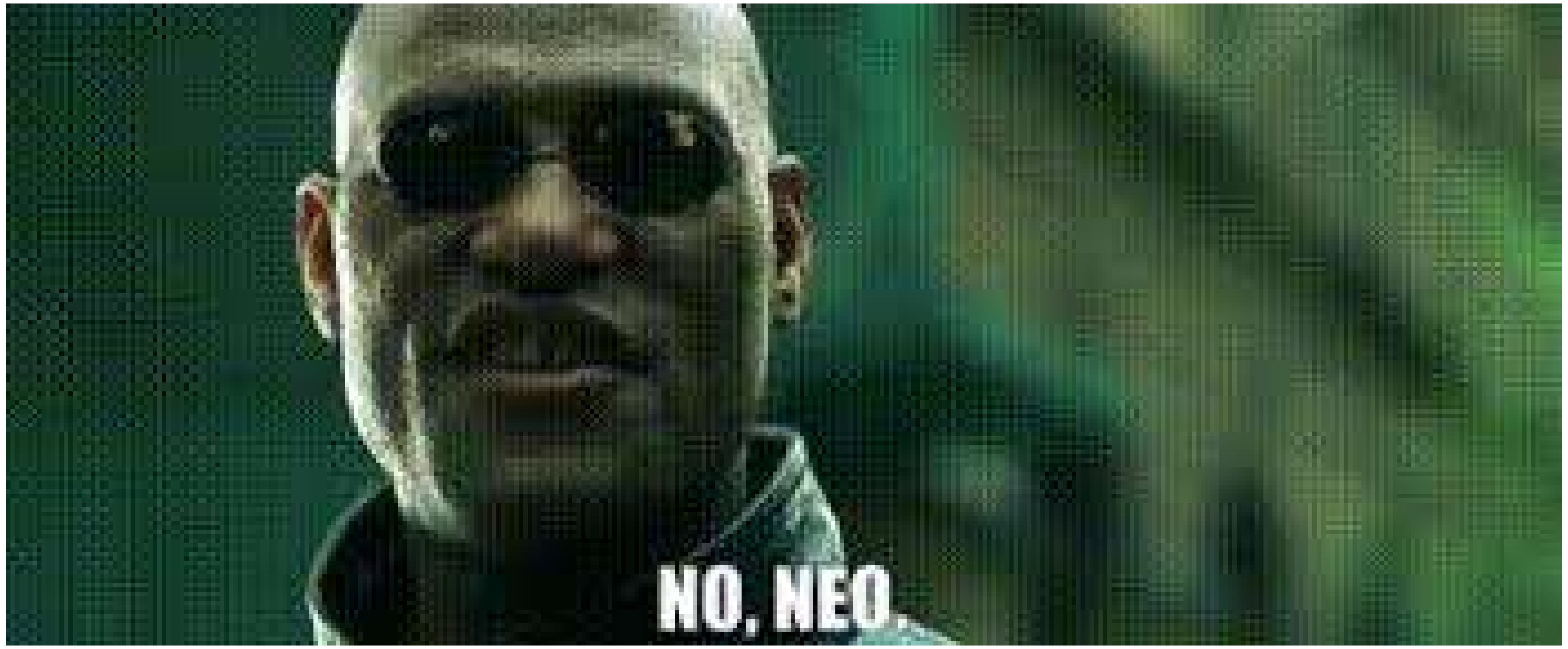
Positive Semi-Definite Matrix

Is Positivity related to the Diagonal?

$$\begin{bmatrix} 1 \\ 1 \end{bmatrix}^\top \begin{bmatrix} 1 & -3 \\ -3 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = -2 \not\geq 0$$

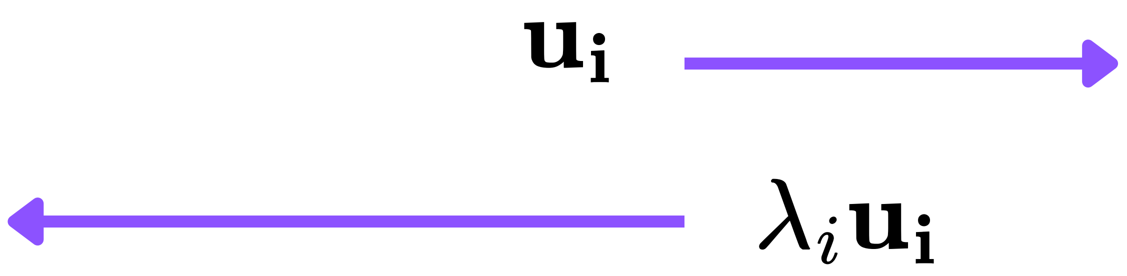
Positive Semi-Definite Matrix

Is Positivity related to the Diagonal?



What else then?

$$\text{Eigenvalues. } \forall \lambda_i : A\mathbf{u}_i = \lambda_i \mathbf{u}_i$$



$$\text{if } \exists \lambda_i < 0, \text{ then } \mathbf{u}_i^\top A\mathbf{u}_i = \lambda_i \mathbf{u}_i^\top \mathbf{u}_i < 0$$

What else then?

Positive nums have **square root**

$$\text{if } A = C^\top C, \text{ then } \forall \mathbf{x} \in \mathbb{R}^n \setminus \{\mathbf{0}\}:$$
$$\mathbf{x}^\top A \mathbf{x} = \mathbf{x}^\top C^\top C \mathbf{x} = \|C\mathbf{x}\|^2 \geq 0$$

Matrices with a square root are positive

Positive Semi-Definite Matrix

Theorem

Let A be a real symmetric $n \times n$ matrix. Then, the following are equivalent:

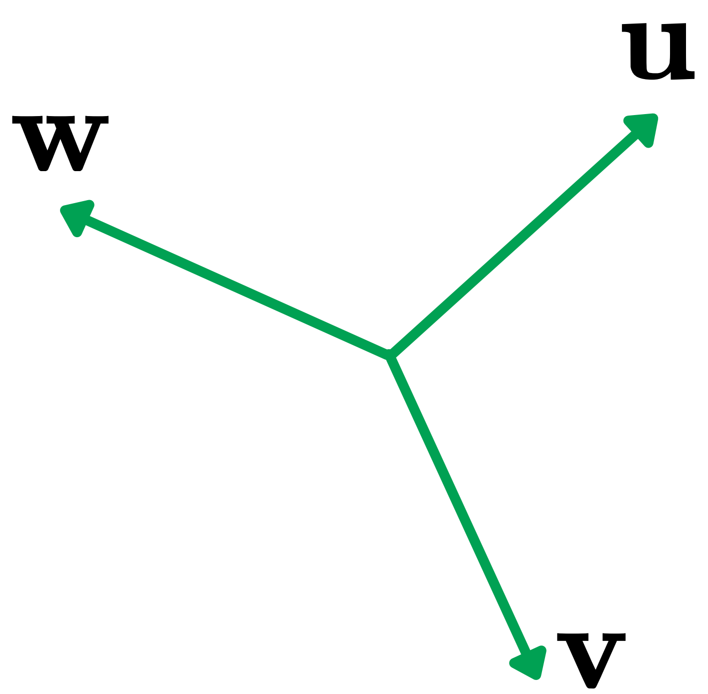
1. $\forall \mathbf{x} \in \mathbb{R}^n, \mathbf{x}^\top A\mathbf{x} \geq 0.$
2. *All eigenvalues of A are nonnegative*
3. *There is an $n \times n$ real matrix C, such that $A = C^\top C$*

THE TRAINING

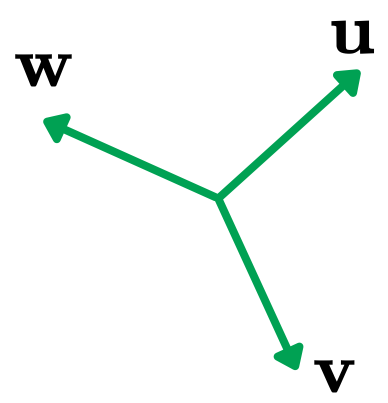
SemiDefinite Prog.

A geometric example

$$\begin{cases} \min & \mathbf{u}^T\mathbf{v} + \mathbf{v}^T\mathbf{w} + \mathbf{w}^T\mathbf{u} \\ \text{s.t.} & \|\mathbf{u}\|_2 = \|\mathbf{v}\|_2 = \|\mathbf{w}\|_2 = 1 \\ & \mathbf{u}, \mathbf{v}, \mathbf{w} \in \mathbb{R}^3 \end{cases}$$



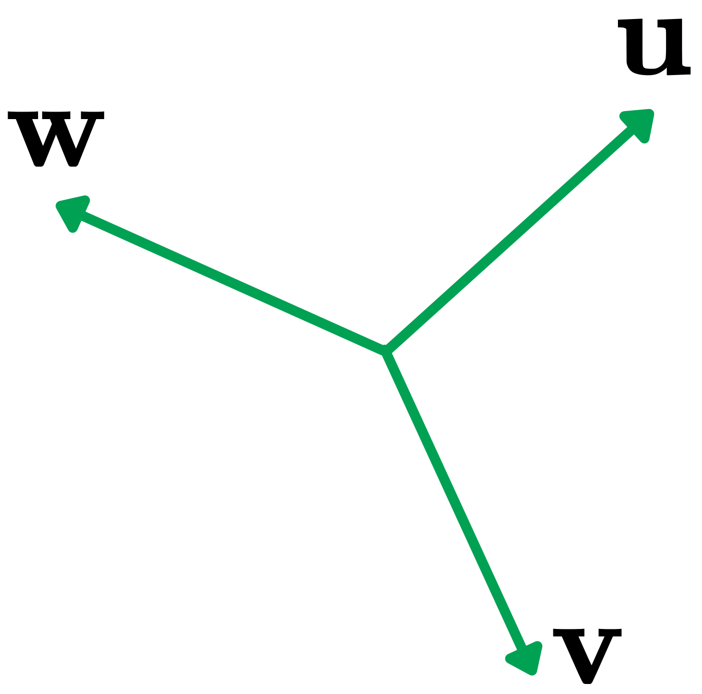
A geometric example



$$\begin{cases} \min & \mathbf{u}^T\mathbf{v} + \mathbf{v}^T\mathbf{w} + \mathbf{w}^T\mathbf{u} \\ \text{s.t.} & \|\mathbf{u}\|_2 = \|\mathbf{v}\|_2 = \|\mathbf{w}\|_2 = 1 \\ & \mathbf{u}, \mathbf{v}, \mathbf{w} \in \mathbb{R}^3 \end{cases}$$

$$\begin{bmatrix} \mathbf{u}^T \\ \mathbf{v}^T \\ \mathbf{w}^T \end{bmatrix} \begin{bmatrix} \mathbf{u} & \mathbf{v} & \mathbf{w} \end{bmatrix} = \begin{bmatrix} 1 & \mathbf{u}^T\mathbf{v} & \mathbf{u}^T\mathbf{w} \\ \mathbf{v}^T\mathbf{u} & 1 & \mathbf{v}^T\mathbf{w} \\ \mathbf{w}^T\mathbf{u} & \mathbf{w}^T\mathbf{v} & 1 \end{bmatrix} = \begin{bmatrix} 1 & a & b \\ a & 1 & c \\ b & c & 1 \end{bmatrix} = A \geq 0$$

A geometric example



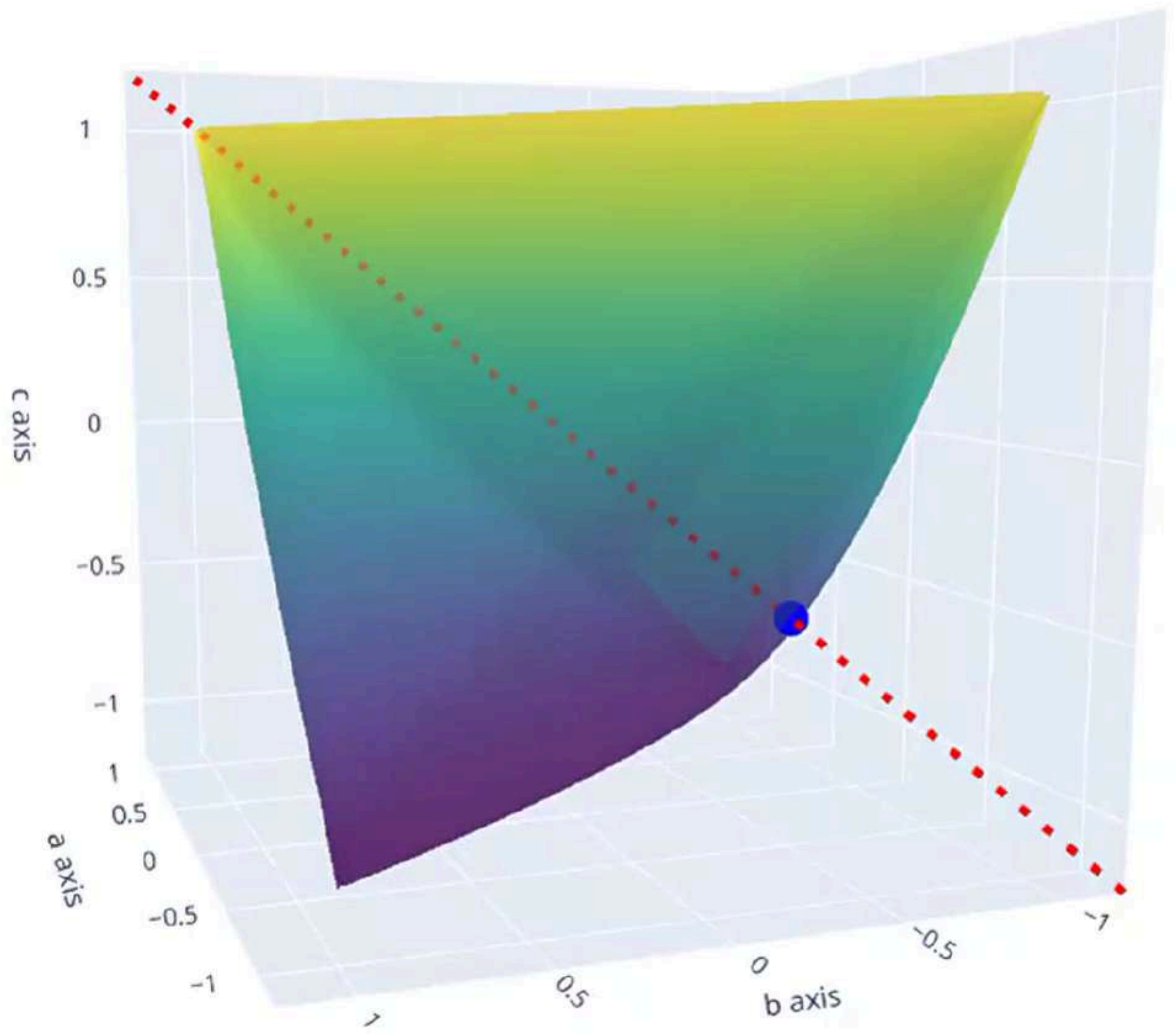
$$\begin{bmatrix} \mathbf{u}^T\mathbf{v} \\ \mathbf{u}^T\mathbf{w} \\ \mathbf{v}^T\mathbf{w} \end{bmatrix} = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$$

SemiDefinite Programming

$$\begin{cases} \min \quad a+b+c \\ \text{s.t.} \quad A = \begin{bmatrix} 1 & a & b \\ a & 1 & c \\ b & c & 1 \end{bmatrix} \geq 0 \end{cases}$$

A geometric example

sol: $a = b = c = -0.5$



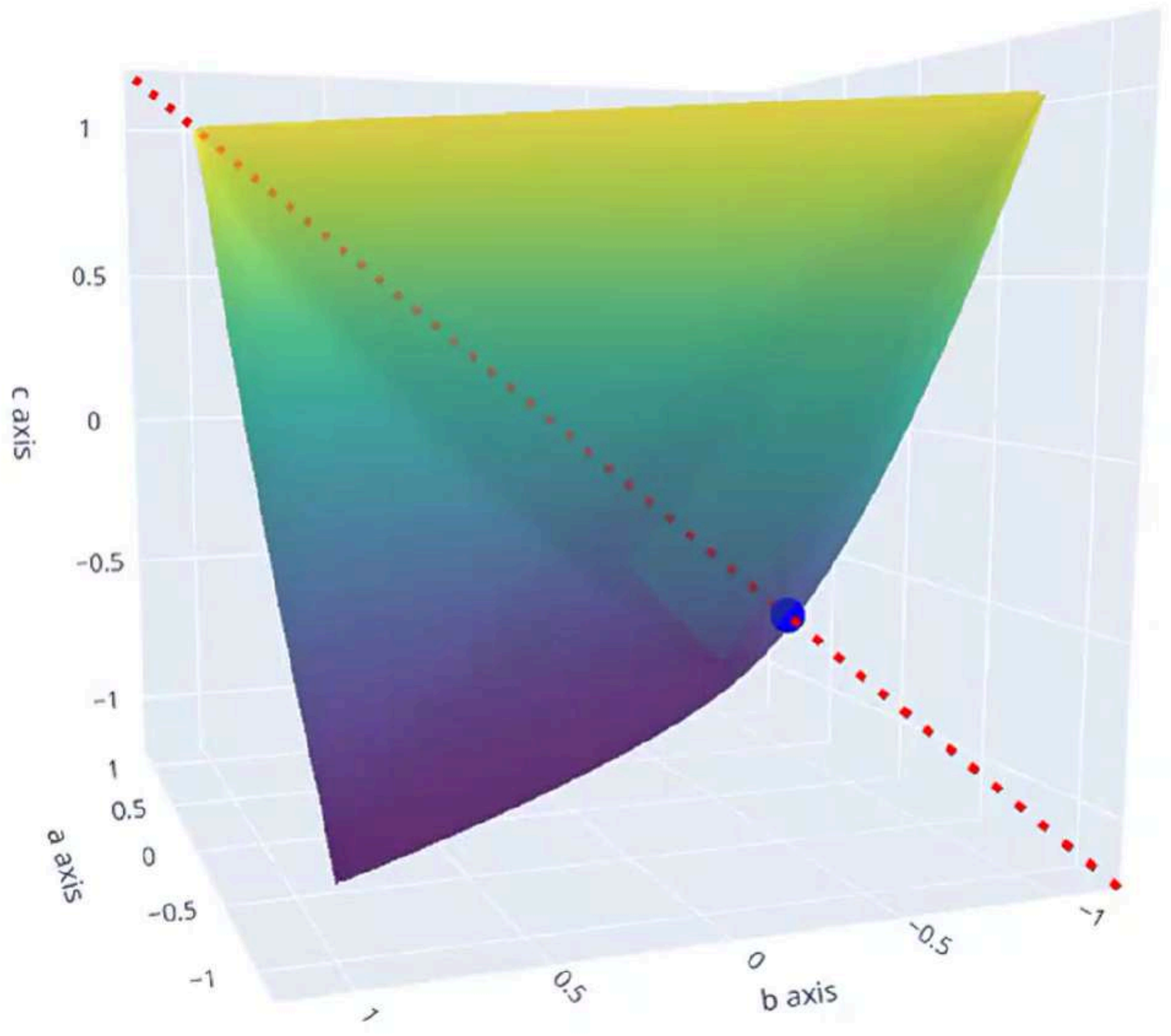
Elliptope

SemiDefinite Programming

$$\begin{cases} \min \quad a+b+c \\ \\ \text{s.t.} \quad A = \begin{bmatrix} 1 & a & b \\ a & 1 & c \\ b & c & 1 \end{bmatrix} \geq 0 \end{cases}$$

A geometric example

sol: $a = b = c = -0.5$



Elliptope

SemiDefinite Programming

$$\begin{cases} \min \quad a + b + c \\ \text{s.t.} \quad A = \begin{bmatrix} 1 & a & b \\ a & 1 & c \\ b & c & 1 \end{bmatrix} \geq 0 \end{cases}$$

Cholesky Decomposition

to get $\mathbf{u}, \mathbf{v}, \mathbf{w}$ back

A geometric example

SemiDefinite Programming

$$\begin{cases} \min \quad a+b+c \\ \text{s.t.} \quad A = \begin{bmatrix} 1 & a & b \\ a & 1 & c \\ b & c & 1 \end{bmatrix} \geq 0 \end{cases}$$

polynomial time complexity, for accuracy fixed to ε: ←

$O(n^k log(1/\epsilon))$

Cholesky Decomposition to get $\mathbf{u}, \mathbf{v}, \mathbf{w}$ back → $O(n^3)$

THE REAL WORLD

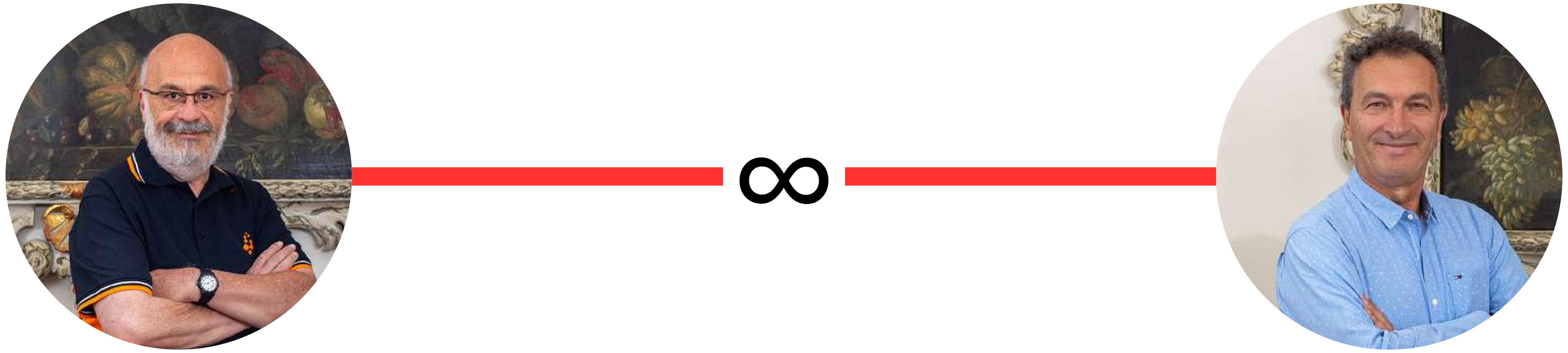
Max-Cut

The real problem

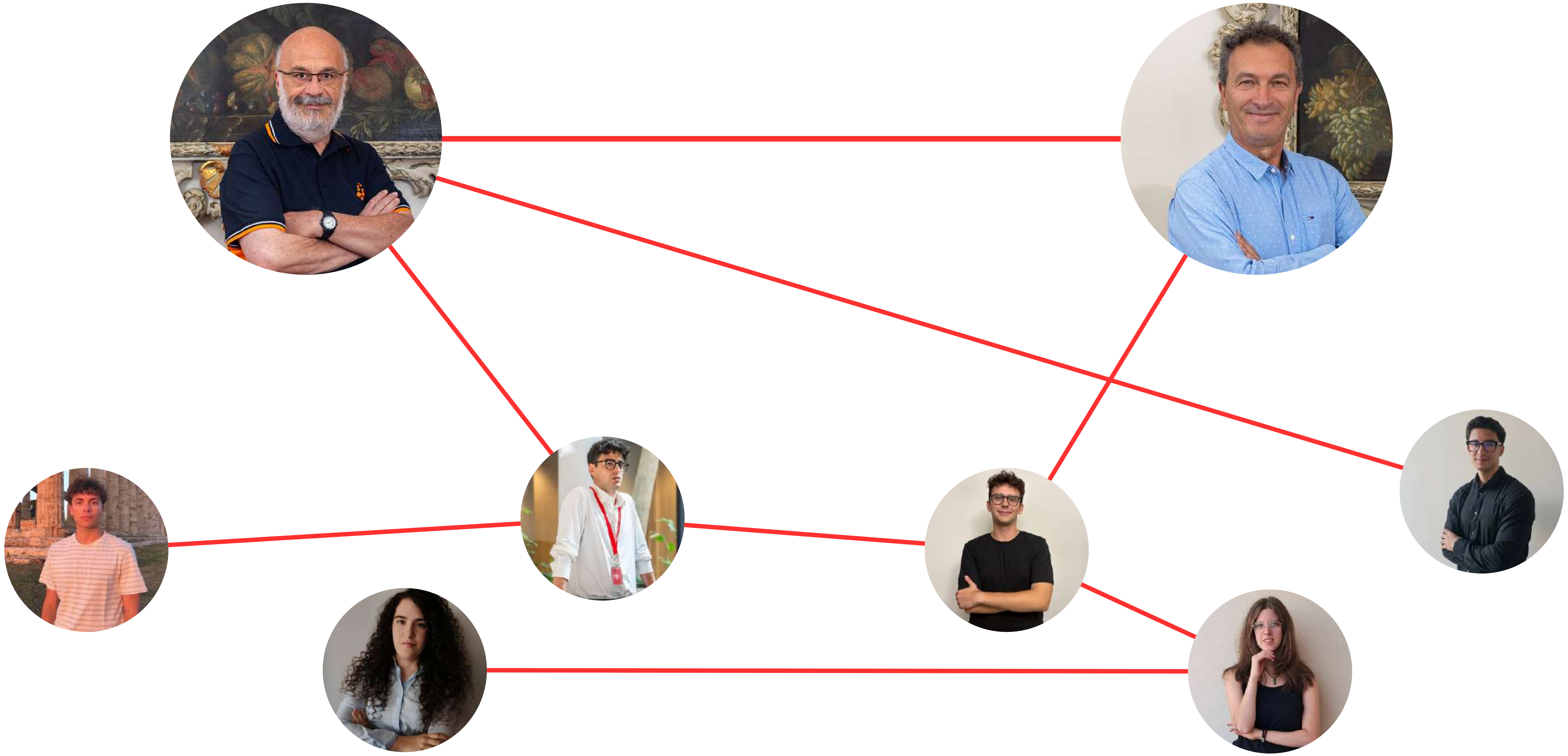
ORTOGONALI

PARALLELI

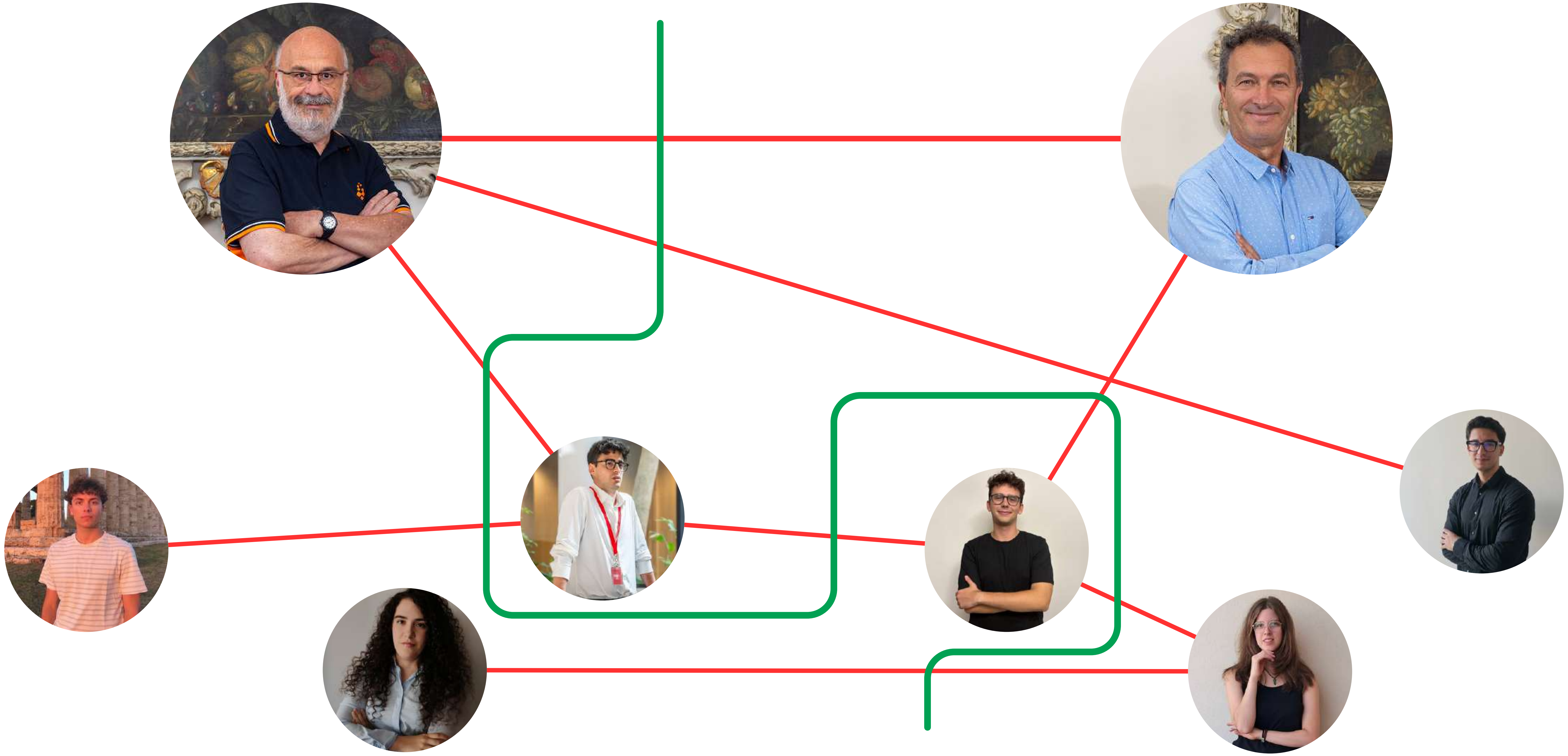
The real problem



The real problem

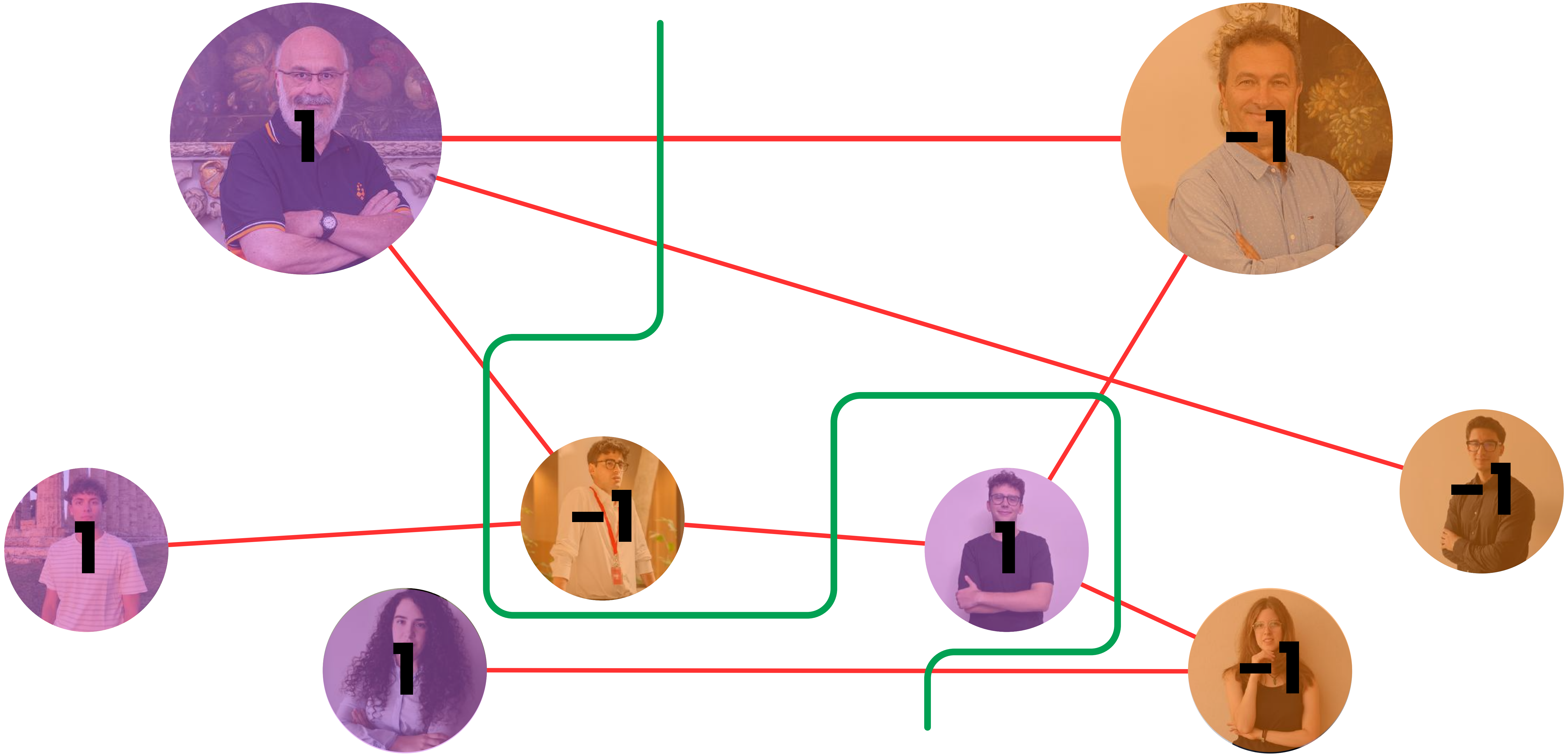


The real problem



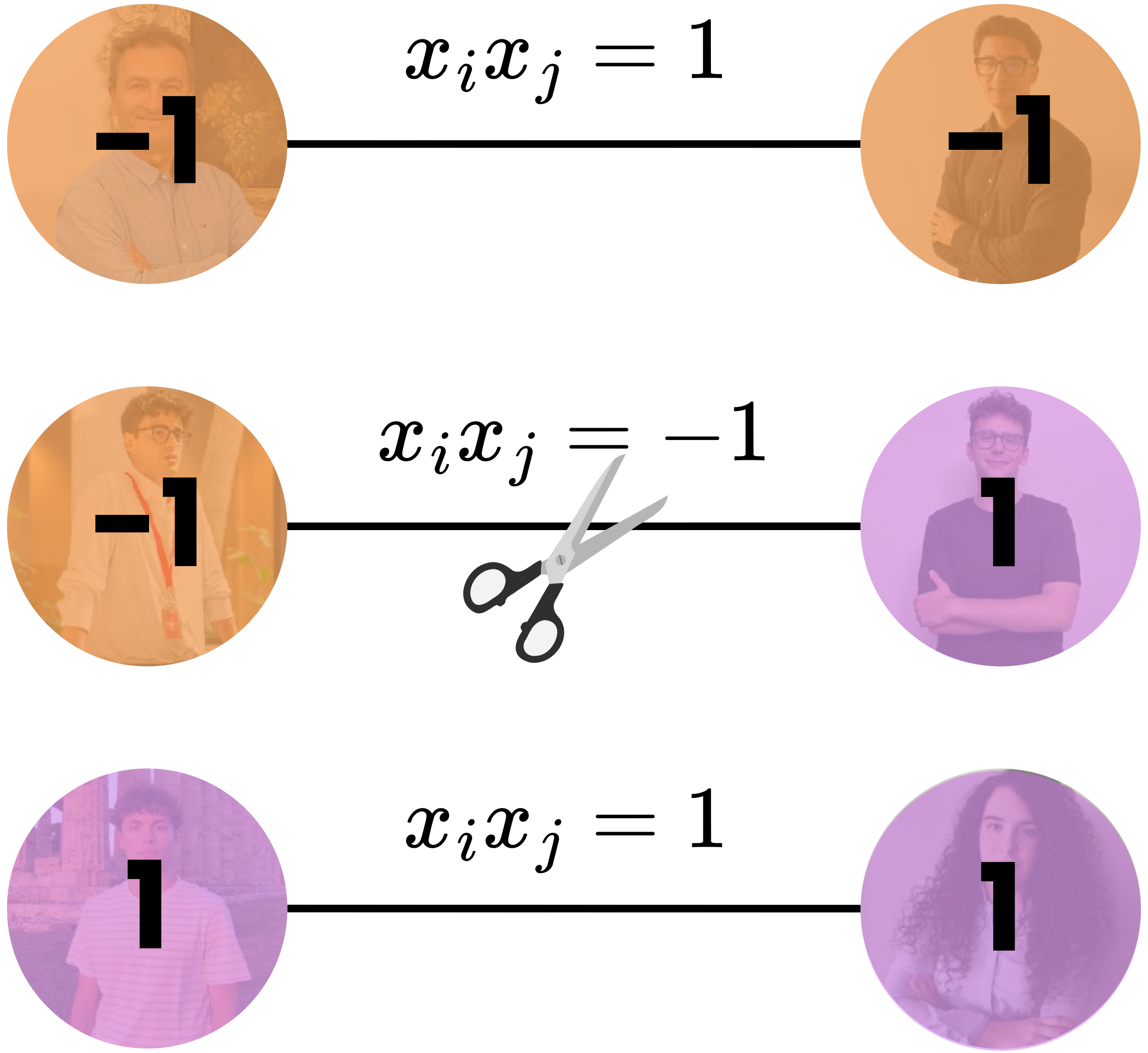
$$cut(S, V - S) = 8$$

The real problem



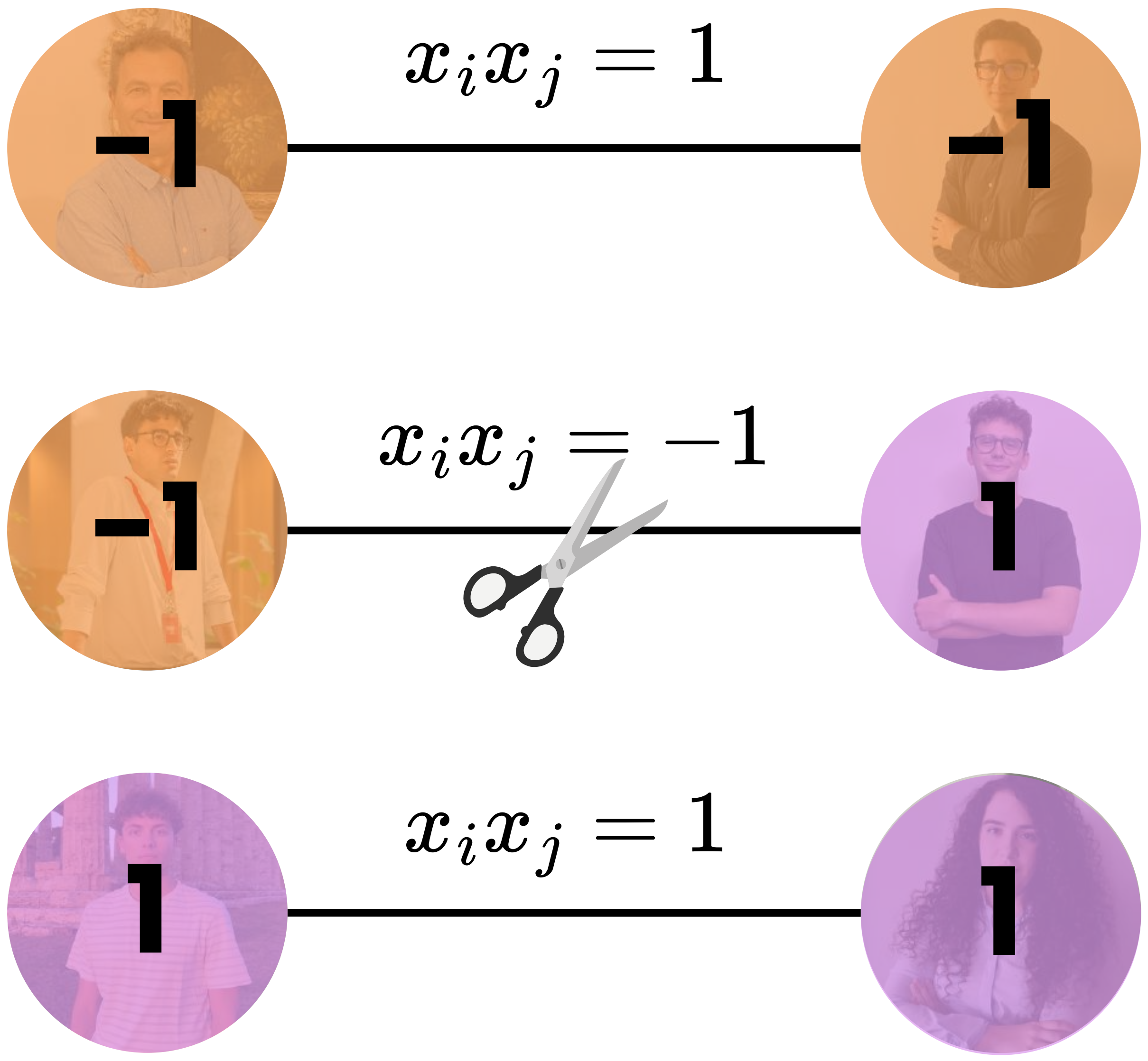
$cut(S, V - S) = 8$

The real problem



$$S = \{all\ v_i \in V\ s.t.\ x_i = 1\}$$

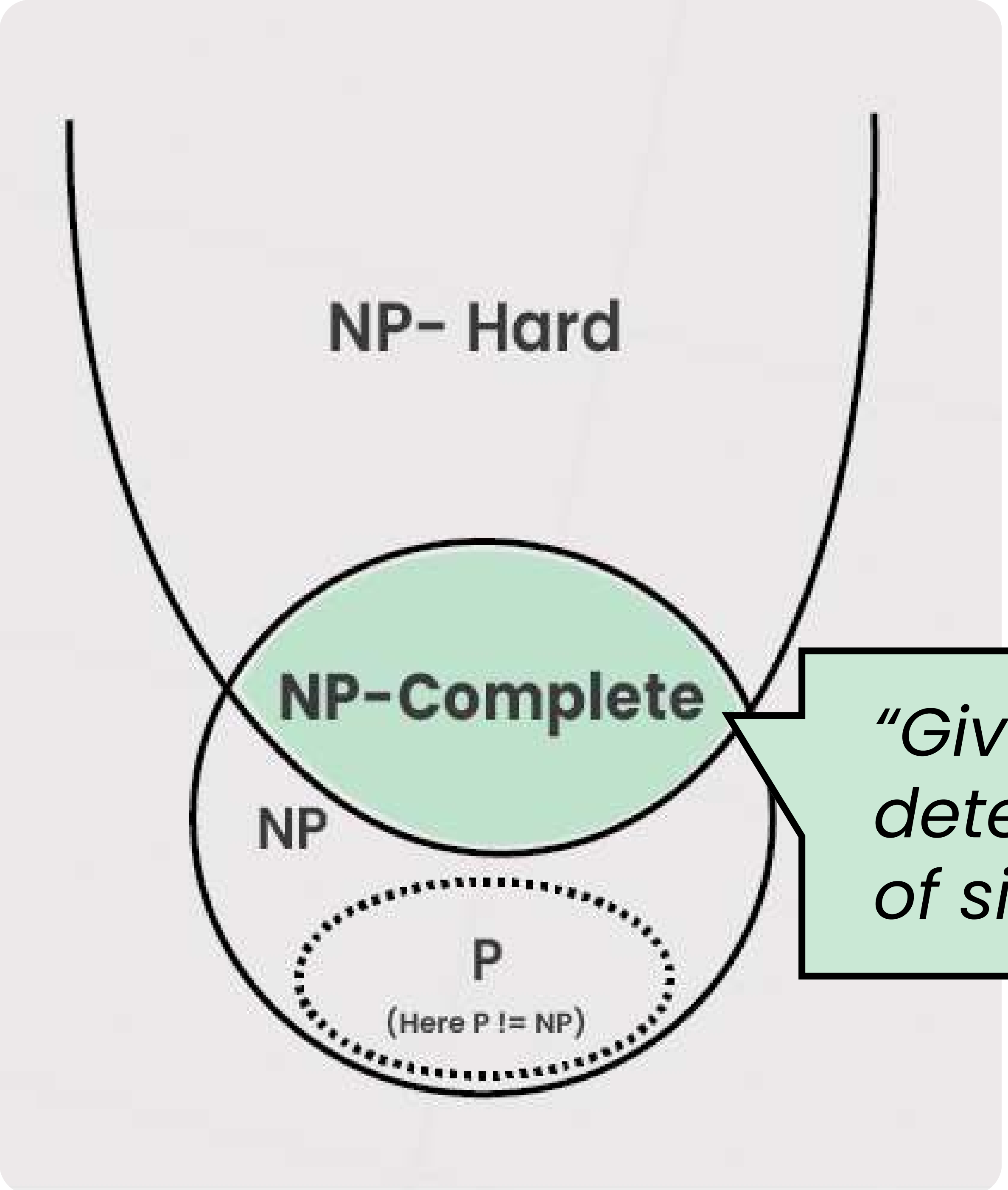
Max-Cut



$$\mathbb{1}[cut_{ij}] = \frac{1 - x_i x_j}{2} = \begin{cases} 1 & if\ x_i \neq x_j \\ 0 & if\ x_i = x_j \end{cases}$$

$$\textbf{Max-Cut} = \max_{x} \sum_{e_{ij} \in E} w_{ij} \frac{1 - x_i x_j}{2}$$

Max-Cut: Complexity



"Given a graph G and an integer k, determine whether there is a cut of size at least k in G."

Max-Cut: Approximation

What can we do in polynomial time?

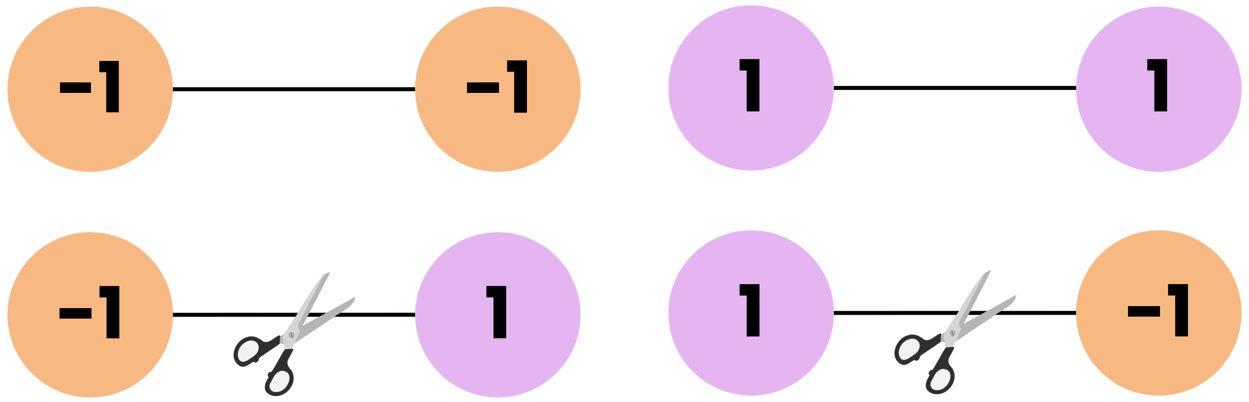
Greedy

$$\forall v_i \in V : \; x_i = \begin{cases} 1 & w.p. \; 0.5 \\ -1 & w.p. \; 0.5 \end{cases}$$

using a fair coin, independent ∀i

$$S = \{v_i \in V \; s.t. \; x_i = 1\}$$

Max-Cut: Approximation



$$\mathbb{E}[cut(S, V - S)] = \frac{|E|}{2}$$

$$\forall S : cut(S, V - S) \leq |E|$$

Greedy is 0.5-approx of Max-Cut

Max-Cut: Approximation

Greedy is 0.5-approx of Max-Cut

Max-Cut: Approximation

$$\begin{cases} \max \sum_{e_{ij} \in E} \frac{1 - x_i x_j}{2} \\ \\ \text{s.t.} \quad x_i^2 = 1 \\ \qquad\quad x_i \in \mathbb{R} \\ \qquad\quad \forall v_i \in V \end{cases}$$

This **QP is NP-Hard** to solve:
search space is **non-convex.**

Max-Cut: Approximation

$$\begin{cases} \max \sum_{e_{ij} \in E} x_{ij} \\ \\ \quad \text{s.t.} \quad 0 \leq x_{ij} \leq 1 \\ x_{ij} + x_{jk} + x_{ki} \leq 2 \\ \qquad\qquad \forall e_{ij} \in E \end{cases}$$

"All LPs are 0.5-approx (or lower) for Max-Cut, provided polynomial size and constant arity."

THE ONE

Goemans-Williamson

Max-Cut: SDP

Relaxation:
use unitary norm **vectors** instead of scalar variables.

Vector Program

$$\begin{cases} \max \frac{1}{2} \sum_{e_{ij} \in E} (1 - \mathbf{x}_i^T \mathbf{x}_j) \\ \\ \text{s.t.} \quad \|\mathbf{x}_i\| = 1 \\ \qquad\quad \mathbf{x}_i \in \mathbb{R}^n \\ \qquad\quad \forall v_i \in V \end{cases}$$

$$\text{Max-Cut} \leq \text{Vec.Prog.-OPT}$$

Max-Cut: SDP

Relaxation:
use unitary norm **vectors** instead of scalar variables.
Then, exploit **SDP**.

1. Define: $X = [\mathbf{x}_i^T \mathbf{x}_j]_{i,j}, \quad \mathbf{x}_i \in \mathbb{S}^{n-1}$

$$X = [\mathbf{x}_1 \dots \mathbf{x}_n]^T [\mathbf{x}_1 \dots \mathbf{x}_n] \geq 0$$

Max-Cut: SDP

Relaxation:
use unitary norm **vectors** instead of scalar variables.
Then, exploit **SDP**.

1. Define: $X = [\mathbf{x}_i^T \mathbf{x}_j]_{i,j}, \quad \mathbf{x}_i \in \mathbb{S}^{n-1}$

$$X = [\mathbf{x}_1 \dots \mathbf{x}_n]^T [\mathbf{x}_1 \dots \mathbf{x}_n] \geq 0$$

2. Solve SDP for X

2.1 SDP-OPT = Vec.Prog.-OPT

2.2 Approx. sol in $O(n^k/\epsilon)$

$$\begin{cases} \max \frac{1}{2} \sum_{e_{ij} \in E} (1 - X_{ij}) \\ \text{s.t.} \quad X \in \mathbb{R}^{n \times n} \\ \qquad X \geq 0 \\ \qquad X_{ii} = 1, \quad i = 1 \dots n \end{cases}$$

Max-Cut: SDP

Relaxation:
use unitary norm **vectors** instead of scalar variables.
Then, exploit **SDP**.

1. Define: $X = [\mathbf{x}_i^T \mathbf{x}_j]_{i,j}, \quad \mathbf{x}_i \in \mathbb{S}^{n-1}$

$$X = [\mathbf{x}_1 \dots \mathbf{x}_n]^T [\mathbf{x}_1 \dots \mathbf{x}_n] \geq 0$$

2. Solve SDP for X

3. Cholesky Decomposition to get $\mathbf{x}_i$

4. Randomized **Rounding**

$$\begin{cases} \max \frac{1}{2} \sum_{e_{ij} \in E} (1 - X_{ij}) \\ \text{s.t.} \quad X \in \mathbb{R}^{n \times n} \\ \qquad X \geq 0 \\ \qquad X_{ii} = 1, \quad i = 1 \dots n \end{cases}$$

Max-Cut: SDP

4. Randomized **Rounding**

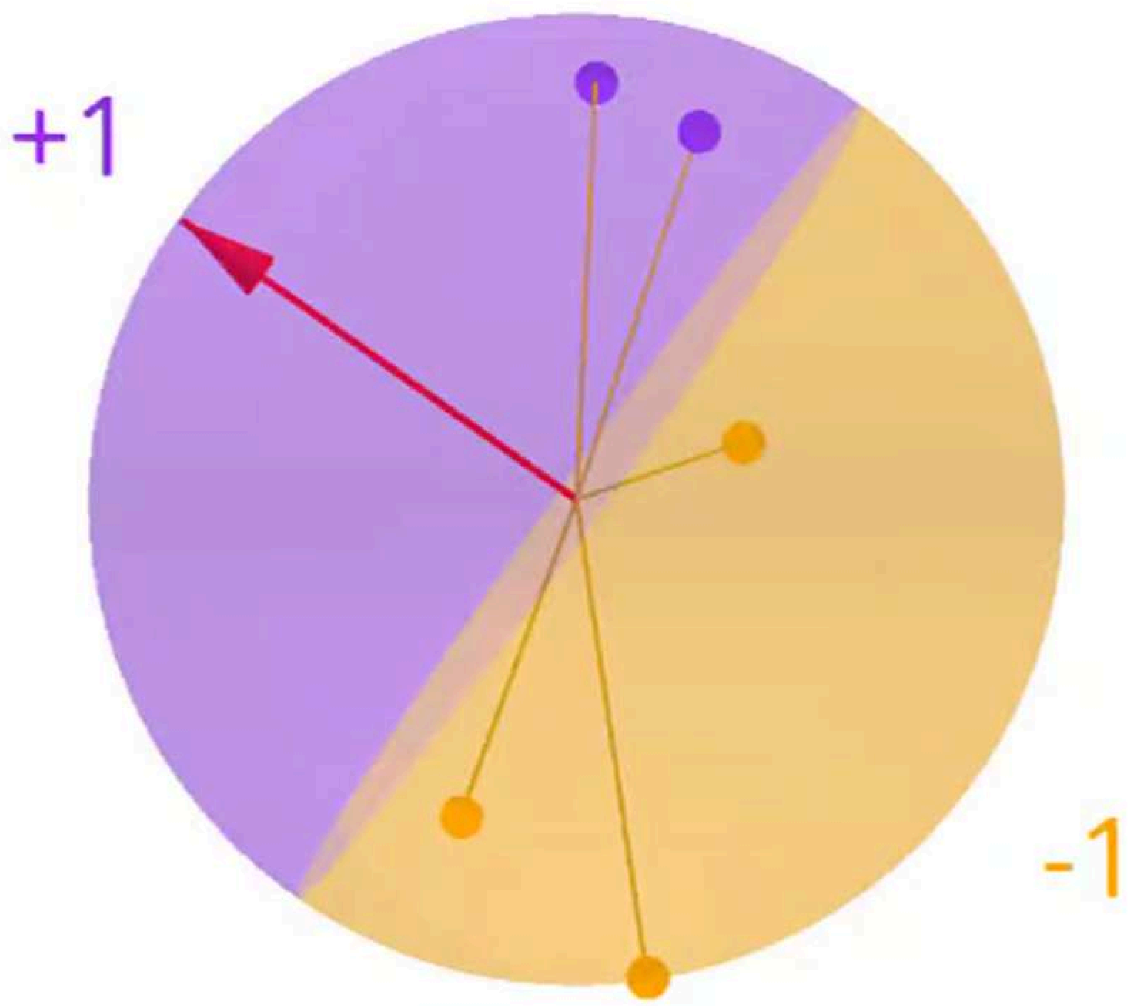
$$\mathbf{x}_i \in \mathbb{S}^{n-1} \to \mathbf{x}_i \in \{-1, 1\}$$

4.1 define a random hyperplane

$$\mathbf{y} \sim \mathcal{N}(0, I)$$

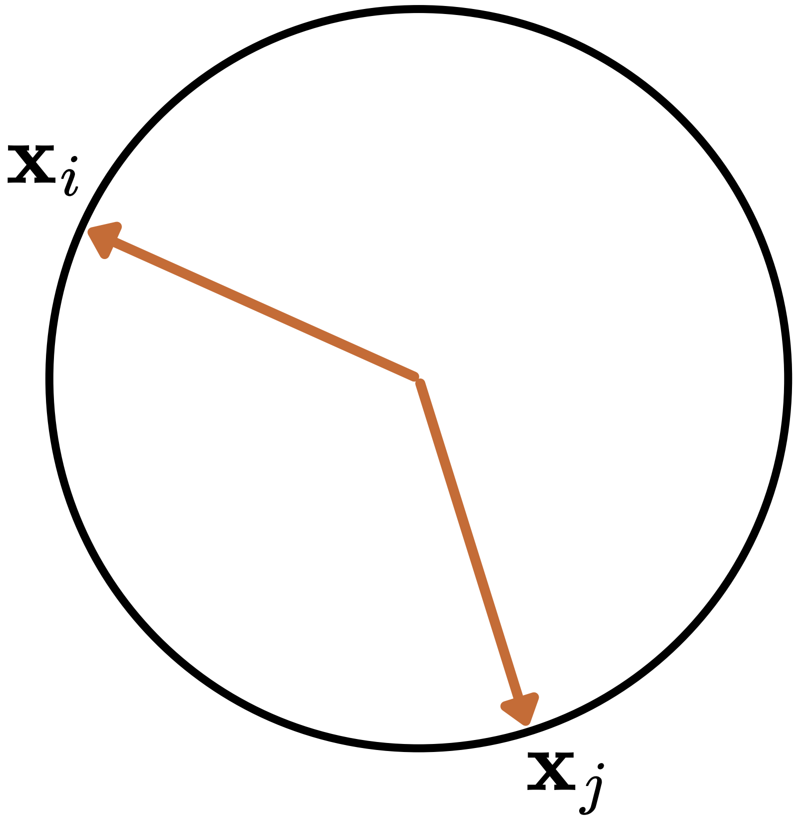
4.2 Assign:

- **-1** if $\mathbf{y}^T \mathbf{x}_i < 0$
- **1** if $\mathbf{y}^T \mathbf{x}_i \geq 0$



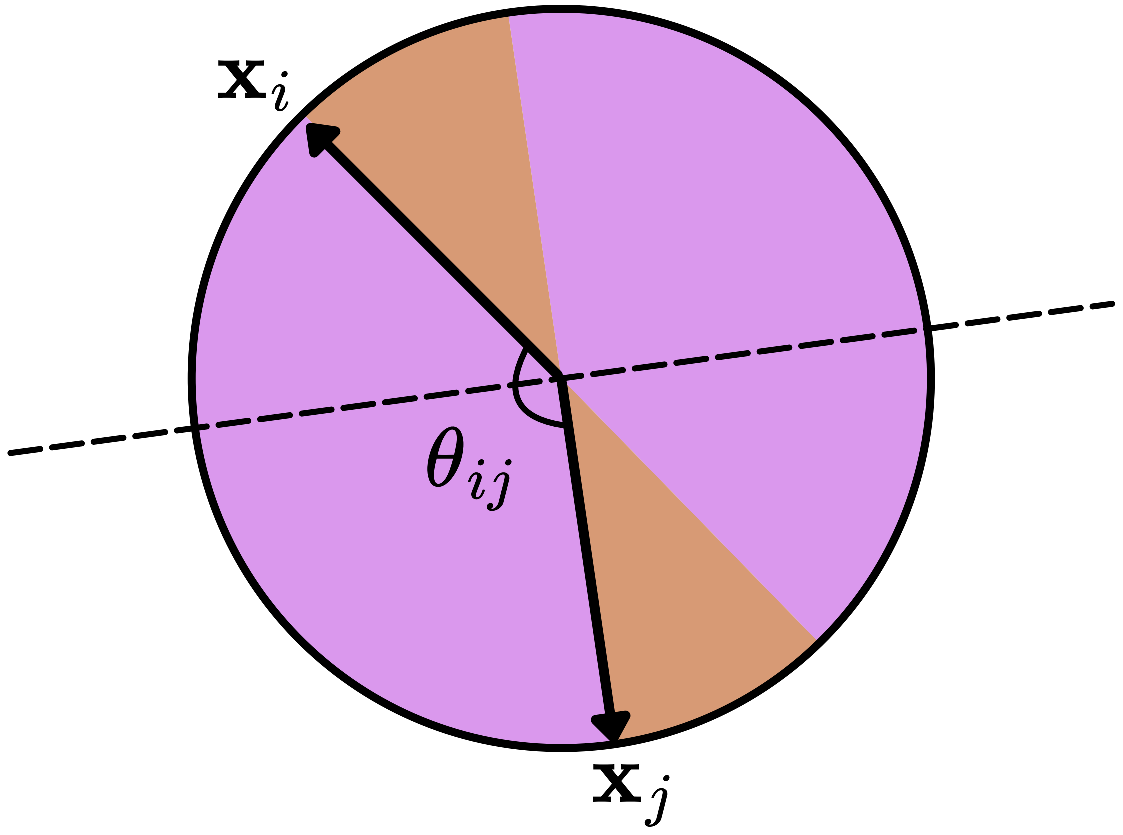
Max-Cut: SDP Analysis

$$\mathbb{P}(label\ x_i \neq label\ x_j) = ?$$



Max-Cut: SDP Analysis

$$\mathbb{P}(label\ x_i \neq label\ x_j) = \frac{\theta_{ij}}{\pi}$$



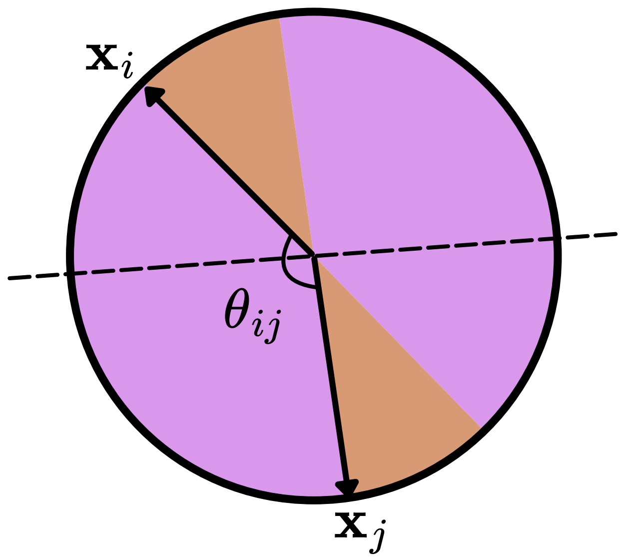
Max-Cut: SDP Analysis

$$\mathbb{E}[cut_{G\&W}] = \sum_{e_{ij}\in E} \frac{\theta_{ij}}{\pi}$$

vs

$$\text{SDP-OPT} = \sum_{e_{ij}\in E} \frac{1-\mathbf{x}_i^T\mathbf{x}_j}{2}$$

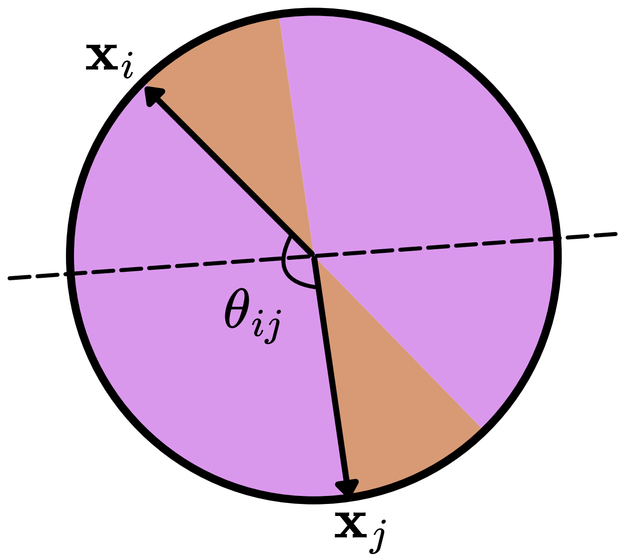
$$\mathbb{P}(cut_{ij}) = \frac{\theta_{ij}}{\pi}$$



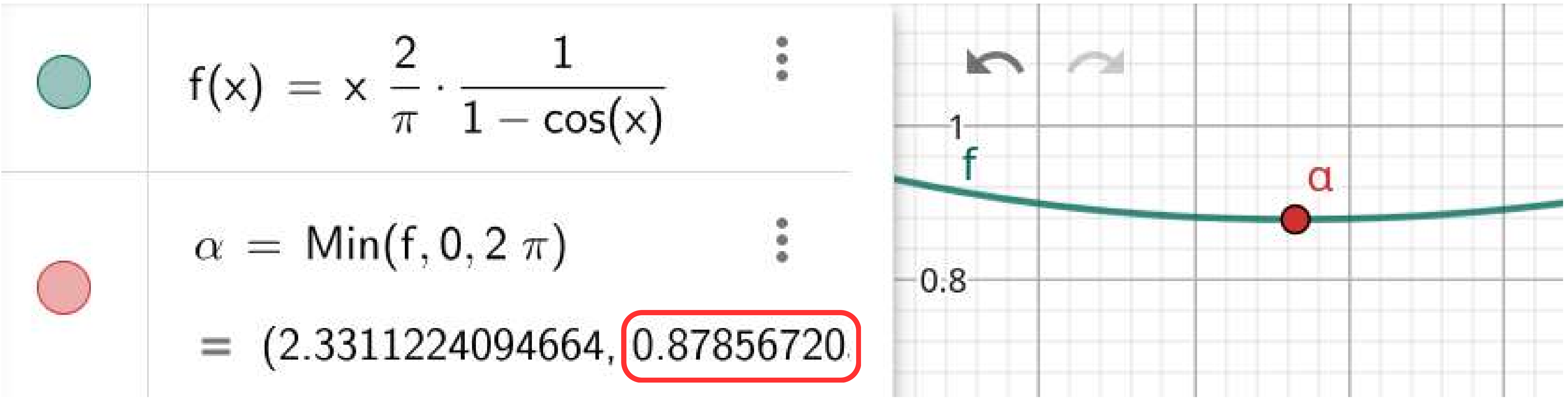
Max-Cut: SDP Analysis

$$\mathbb{E}[cut_{G\&W}] = \sum_{e_{ij} \in E} \frac{\theta_{ij}}{\pi} \qquad \textit{vs} \qquad \text{SDP-OPT} = \sum_{e_{ij} \in E} \frac{1 - \mathbf{x}_i^T \mathbf{x}_j}{2}$$

$$\mathbb{P}(cut_{ij}) = \frac{\theta_{ij}}{\pi}$$



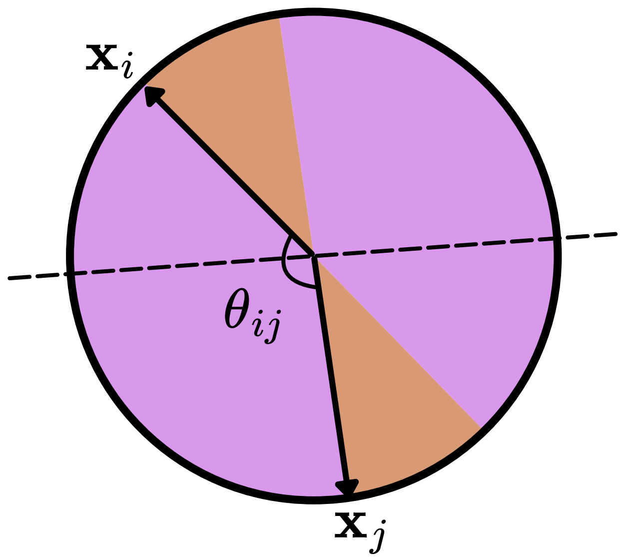
$$\frac{G\&W}{\text{SDP-OPT}} = \frac{\theta_{ij}}{\pi} \left(\frac{1 - \mathbf{x}_i^T \mathbf{x}_j}{2} \right)^{-1} = \frac{\theta_{ij}}{\pi} \left(\frac{1 - cos(\theta_{ij})}{2} \right)^{-1} \geq 0.87\ldots \quad := \alpha$$



Max-Cut: SDP Analysis

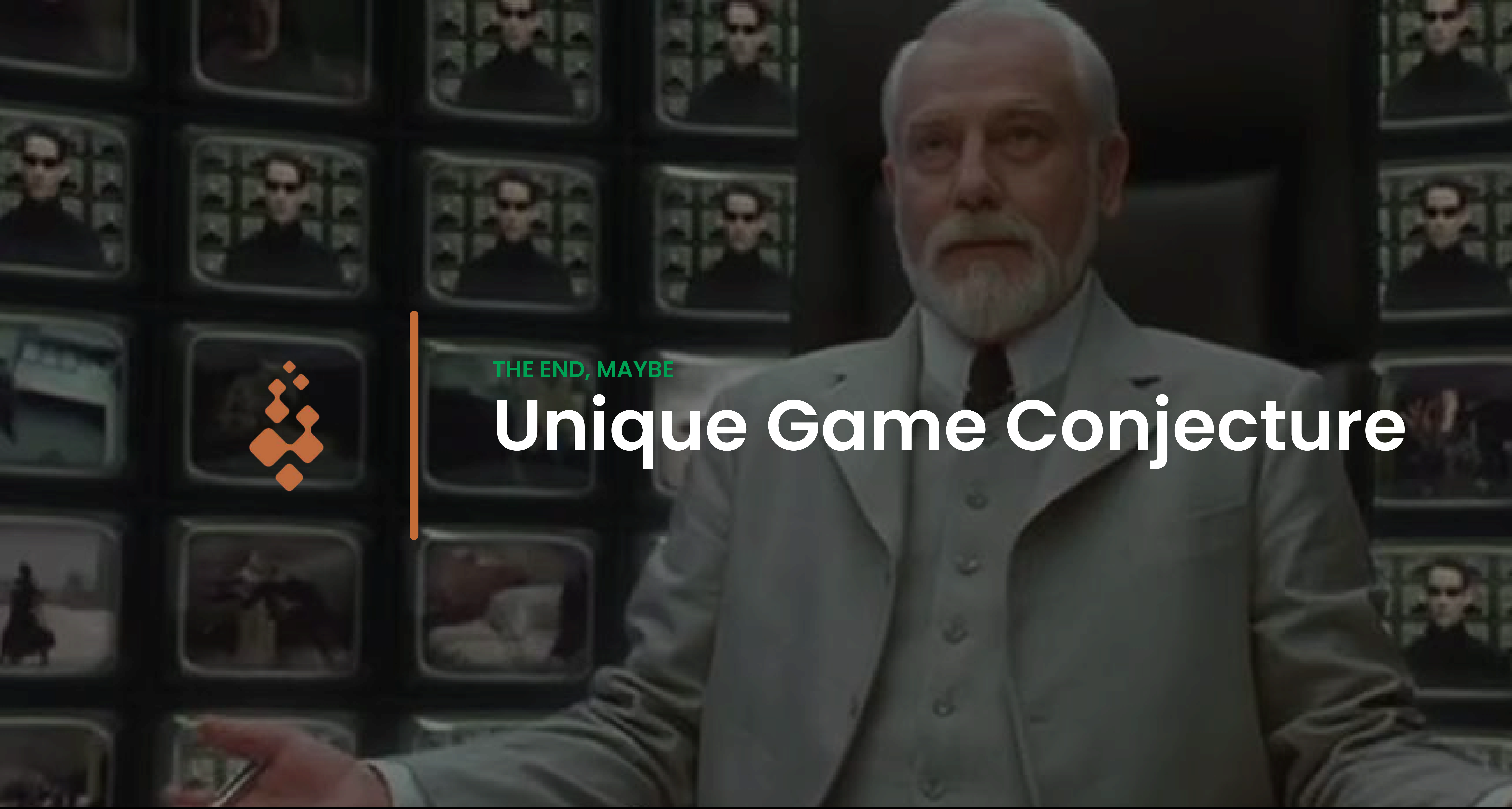
$$\mathbb{E}[cut_{G\&W}] = \sum_{e_{ij} \in E} \frac{\theta_{ij}}{\pi} \qquad \textbf{vs} \qquad \text{SDP-OPT} = \sum_{e_{ij} \in E} \frac{1 - \mathbf{x}_i^T \mathbf{x}_j}{2}$$

$$\mathbb{P}(cut_{ij}) = \frac{\theta_{ij}}{\pi}$$



$$\frac{G\&W}{\text{SDP-OPT}} = \frac{\theta_{ij}}{\pi}\left(\frac{1 - \mathbf{x}_i^T \mathbf{x}_j}{2}\right)^{-1} = \frac{\theta_{ij}}{\pi}\left(\frac{1 - cos(\theta_{ij})}{2}\right)^{-1} \geq 0.87\ldots \quad := \alpha$$

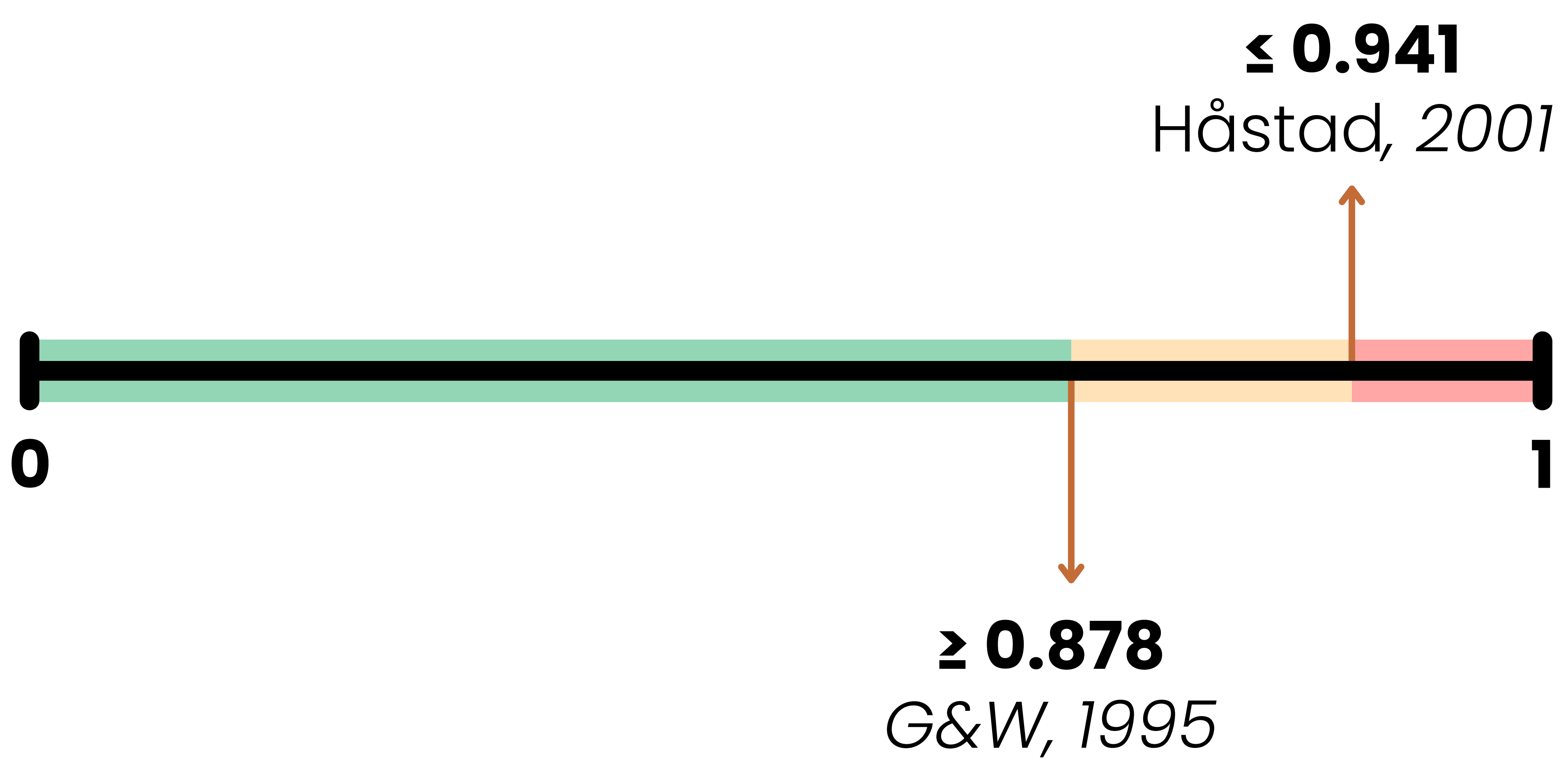
$$\mathbb{E}[cut_{G\&W}] = \sum_{e_{ij} \in E} \frac{\theta_{ij}}{\pi} \geq \alpha \sum_{e_{ij} \in E} \frac{1 - \mathbf{x}_i^T \mathbf{x}_j}{2} = \alpha \text{ SDP-OPT} \geq \alpha \text{ Max-Cut}$$



THE END, MAYBE

Unique Game Conjecture

What can we do in poly time?



Constraint Satisfaction Problem

Max-Cut CSP Definition

Variables:	$V = \{v_1, v_2, \dots, v_n\}$
Domain:	$\mathcal{D} = \{-1, 1\}$
Constraints:	$\mathcal{C} = \{c_{ij} : (i,j) \in E\}$
Constraint Type:	$c_{ij}(x_i, x_j) = \begin{cases} 1 & \text{if } x_i \neq x_j \\ 0 & \text{if } x_i = x_j \end{cases}$
Objective:	$\max_{x \in \mathcal{D}^n} \sum_{(i,j) \in E} w_{ij} \cdot c_{ij}(x_i, x_j)$

Unique Games

Unique Games CSP Definition

Variables:	$V = \{x_1, x_2, \dots, x_n\}$
Domain/Alphabet:	$\Sigma = \{1, 2, \dots, k\}$
Constraints:	$\mathcal{C} = \{\pi_{ij} : (i,j) \in E\}$
Constraint Type:	$\pi_{ij} : \Sigma \to \Sigma$ (A bijection/permutation)
Satisfaction:	Constraint π_{ij} is satisfied if $x_j = \pi_{ij}(x_i)$
Objective:	$\max_{x \in \Sigma^n} \frac{1}{\lvert E \rvert} \sum_{(i,j) \in E} \mathbb{1}[x_j = \pi_{ij}(x_i)]$

Unique Games Conjecture

Khot, 2002 : ***UGC***

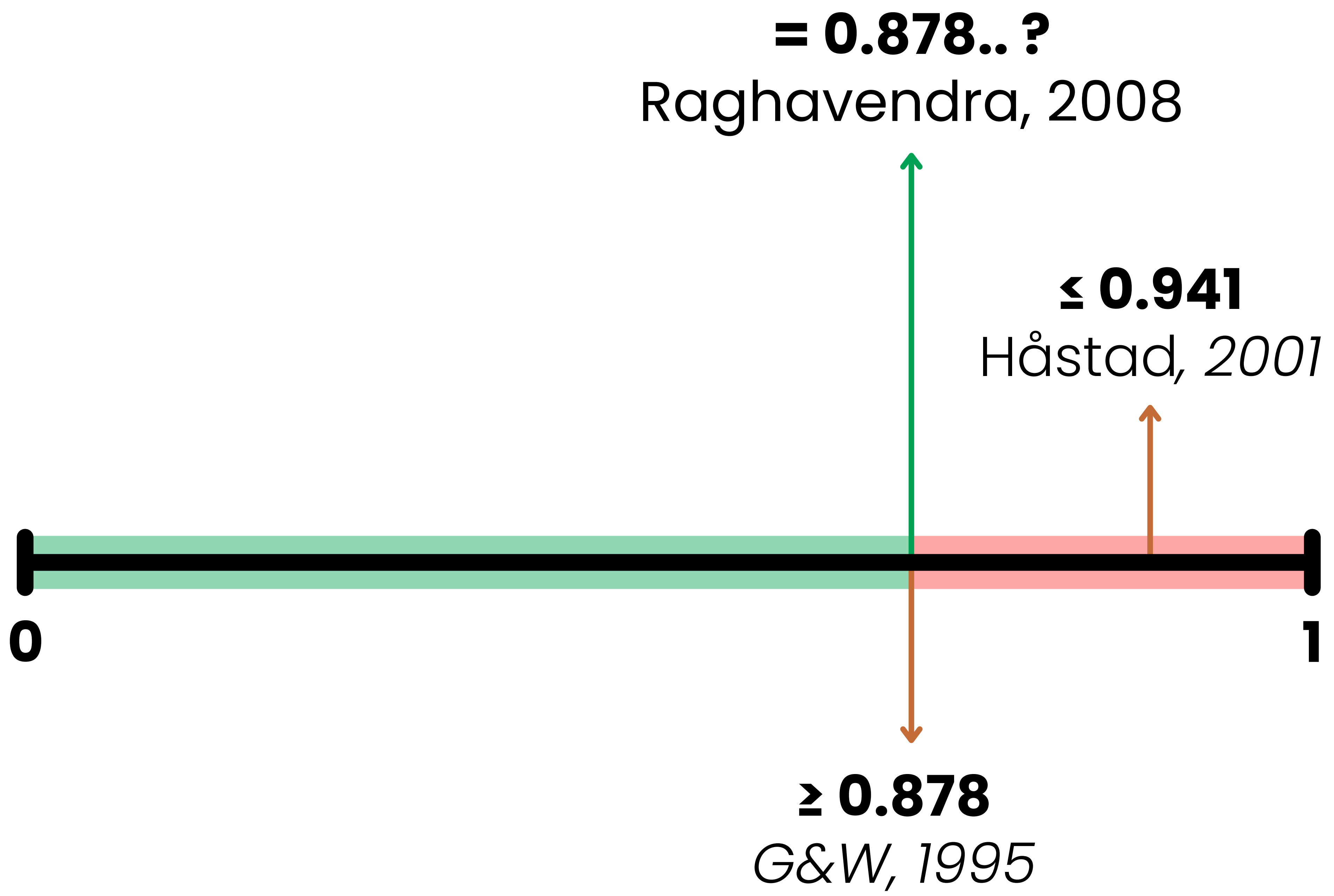
*It's **NP-HARD** to **distinguish** between:*

- *highly satisfiable UG*
- *poorly satisfiable UG*

Raghavendra, 2008

If UGC is true,
***SDP** algos provide the **best approx** ratio for any CSP*

What can we do in poly time?



QUESTIONS?

Thanks for the Attention

References

- Vazirani, V.V., 2001. Approximation algorithms (Vol. 1, Chapt. 26). Berlin: springer.
- Goemans, Michel X., and David P. Williamson. "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming." Journal of the ACM (JACM) 42.6 (1995): 1115-1145.
- Khot, Subhash. "On the power of unique 2-prover 1-round games." Proceedings of the thiry-fourth annual ACM symposium on Theory of computing. 2002.
- Raghavendra, Prasad. "Optimal algorithms and inapproximability results for every CSP?." Proceedings of the fortieth annual ACM symposium on Theory of computing. 2008.
- Håstad, Johan. "Some optimal inapproximability results." Journal of the ACM (JACM) 48.4 (2001): 798-859.