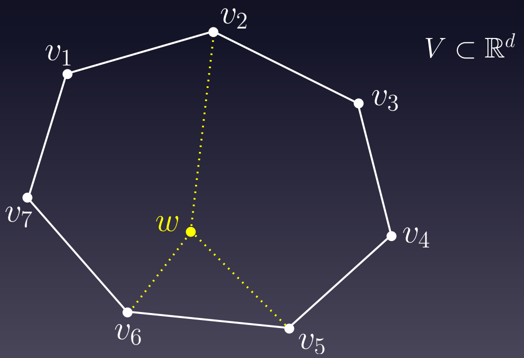
Approximating Nash Equilibria via an Approximate Version of Carathéodory's Theorem

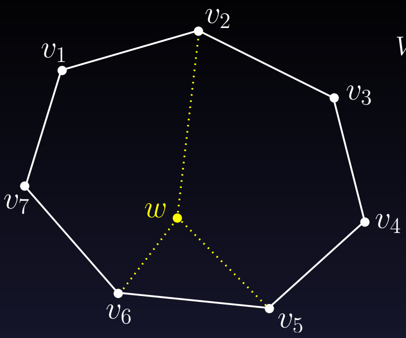
Siddharth Barman

Indian Institute of Science

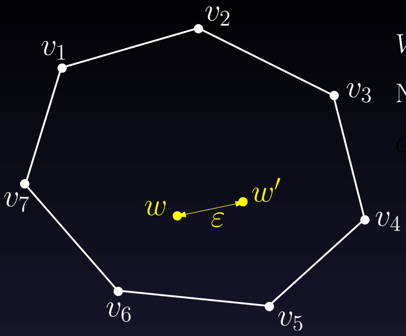
Carathéodory's Theorem

Any vector in the convex hull of a set V in $\mathbb{R}^d$ can be expressed as a convex combination of at most $d+1$ vectors of V.





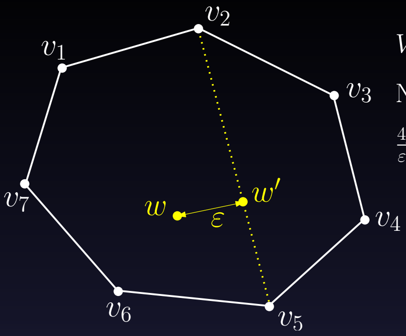
$V \subset \mathbb{R}^d$



$V \subset \mathbb{R}^d$

Norm $p \geq 2$

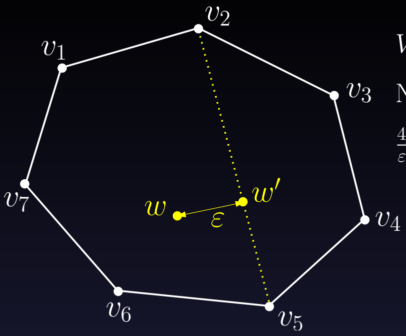
$O\left(\frac{p}{\varepsilon^2}\right)$ vectors



$V \subset \mathbb{R}^d$

Norm $p \geq 2$

$\frac{4p}{\varepsilon^2}$vectors



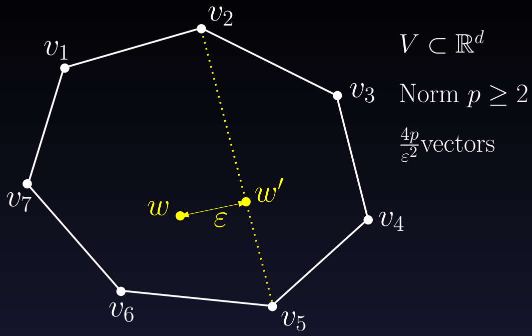
$V \subset \mathbb{R}^d$

Norm $p \geq 2$

$\frac{4p}{\varepsilon^2}$vectors

Approx. Carathéodory's Theorem

Given set V in the p-unit ball with norm $p \geq 2$, for every vector in the convex hull of V there exists an ε-close (under p-norm distance) vector that is a convex combination of at most $\frac{4p}{\varepsilon^2}$ vectors of V.



Approx. Carathéodory's Theorem

Given set V in the p-unit ball with norm $p \geq 2$, for every vector in the convex hull of V there exists an ε-close (under p-norm distance) vector that is a convex combination of at most $\frac{4p}{\varepsilon^2}$ vectors of V.

Proof: Instantiating Maurey's Lemma.
Alternatively, via Khintchine inequality.

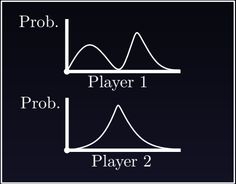
Application I: Approximating Nash Equilibria

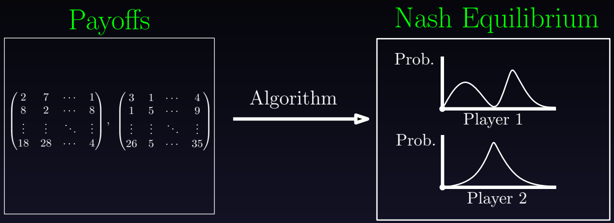
Payoffs

$$\begin{pmatrix} 2 & 7 & \cdots & 1 \\ 8 & 2 & \cdots & 8 \\ \vdots & \vdots & \ddots & \vdots \\ 18 & 28 & \cdots & 4 \end{pmatrix}, \begin{pmatrix} 3 & 1 & \cdots & 4 \\ 1 & 5 & \cdots & 9 \\ \vdots & \vdots & \ddots & \vdots \\ 26 & 5 & \cdots & 35 \end{pmatrix}$$

Algorithm $\longrightarrow$

Nash Equilibrium





Nash equilibrium in two-player games is PPAD-hard [GP06, DGP06, CD06, CDT09].

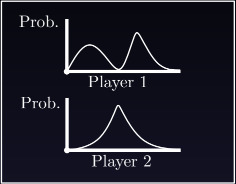
Payoffs

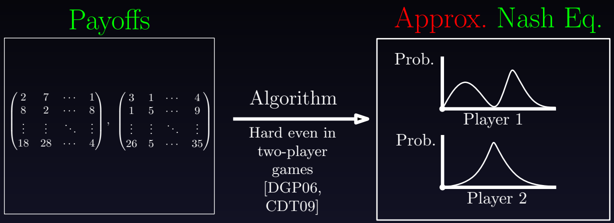
$$\begin{pmatrix} 2 & 7 & \cdots & 1 \\ 8 & 2 & \cdots & 8 \\ \vdots & \vdots & \ddots & \vdots \\ 18 & 28 & \cdots & 4 \end{pmatrix}, \begin{pmatrix} 3 & 1 & \cdots & 4 \\ 1 & 5 & \cdots & 9 \\ \vdots & \vdots & \ddots & \vdots \\ 26 & 5 & \cdots & 35 \end{pmatrix}$$

Algorithm →

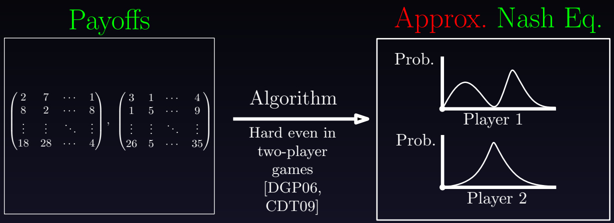
Hard even in two-player games [DGP06, CDT09]

Nash Equilibrium





Focus: Two-Player Games



Focus: Two-Player Games

Two-Player Games model settings in which two self-interested entities *simultaneously* select actions to maximize their own payoffs.

Payoff matrices A and B of size $n \times n$

$$\begin{array}{c c} & \begin{array}{cccc} 1 & 2 & \cdots & n \end{array} \\ \begin{array}{c} 1 \\ 2 \\ \vdots \\ n \end{array} & \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nn} \end{pmatrix} \end{array} \qquad \begin{array}{c c} & \begin{array}{cccc} 1 & 2 & \cdots & n \end{array} \\ \begin{array}{c} 1 \\ 2 \\ \vdots \\ n \end{array} & \begin{pmatrix} B_{11} & B_{12} & \cdots & B_{1n} \\ B_{21} & B_{22} & \cdots & B_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ B_{n1} & B_{n2} & \cdots & B_{nn} \end{pmatrix} \end{array}$$

Payoff matrices A and B of size $n \times n$

Probability vectors over $[n]$: x and y

Payoff matrices A and B of size $n \times n$

Probability vectors over $[n]$: x and y

Nash equilibrium (x, y): No player can benefit by unilateral deviation

$$\begin{aligned} e_i^T A y \leq x^T A y \qquad & \forall i \in [n] \ \text{ and} \\ x^T B e_j \leq x^T B y \qquad & \forall j \in [n] \end{aligned}$$

Payoff matrices A and B of size $n \times n$

Probability vectors over $[n]$: x and y

Nash equilibrium (x, y): No player can benefit by unilateral deviation

$$e_i^T A y \leq x^T A y \qquad \forall i \in [n] \ \text{ and}$$
$$x^T B e_j \leq x^T B y \qquad \forall j \in [n]$$

Approximate Nash equilibrium (x, y): No player can benefit more than ε by unilateral deviation

$$e_i^T A y \leq x^T A y + \varepsilon \qquad \forall i \in [n] \ \text{ and}$$
$$x^T B e_j \leq x^T B y + \varepsilon \qquad \forall j \in [n]$$

Computation of Eq. in Two-Player Games

Nash Equilibria

General Games: Exp. time
[Lemke & Howson 1964]

Zero-Sum Games: Poly. time
[von Neumann 1928, Dantzig 1951]

Computation of Eq. in Two-Player Games

Nash Equilibria

General Games: Exp. time
[Lemke & Howson 1964]

Zero-Sum Games: Poly. time
[von Neumann 1928, Dantzig 1951]

Approximate Nash Equilibria

General Games: $n^{O\left(\log n/\varepsilon^2\right)}$
[Lipton et al. 2003]

Low-Rank Games: $(1/\varepsilon)^{\mathrm{rank}}$
[Alon et al. 2013]

Computation of Eq. in Two-Player Games

Nash Equilibria

General Games: Exp. time
[Lemke & Howson 1964]

Zero-Sum Games: Poly. time
[von Neumann 1928, Dantzig 1951]

Approximate Nash Equilibria

General Games: $n^{O\left(\log n/\varepsilon^2\right)}$
[Lipton et al. 2003]

Low-Rank Games: $(1/\varepsilon)^{\text{rank}}$
[Alon et al. 2013]

This Talk: Sparsity

Definition (Sparsity of a Game)

The sparsity of a game (A, B) is defined to be the maximum number of non-zero entries in any column of $A + B$.

Definition (Sparsity of a Game)

The sparsity of a game (A, B) is defined to be the maximum number of non-zero entries in any column of $A + B$.

- Sparsity $= 0$ in zero-sum games
- In general, sparsity is at most n

Definition (Sparsity of a Game)

The sparsity of a game (A, B) is defined to be the maximum number of non-zero entries in any column of $A + B$.

Theorem

In a two-player s-sparse game an ε-Nash equilibrium can be computed in time $n^{O\left(\log s/\varepsilon^2\right)}$.

Payoff matrices normalized $A, B \in [-1, 1]^{n \times n}$.

Definition (Sparsity of a Game)

The sparsity of a game (A, B) is defined to be the maximum number of non-zero entries in any column of $A + B$.

Theorem

In a two-player s-sparse game an ε-Nash equilibrium can be computed in time $n^{O\left(\log s/\varepsilon^2\right)}$.

Implications:

- When s is a fixed constant we get a polynomial-time algorithm
- For general games $(s \leq n)$ the running time matches the best-known upper bound: $n^{O(\log n/\varepsilon^2)}$ [LMM'03].

Nash eq: $e_i^T Ay \le x^T Ay \quad \forall i$ and

$$x^T Be_j \le x^T By \quad \forall j$$

Bilinear Program for Nash Eq. [MS'64]

$$\begin{array}{ll} \text{maximize} & x^T(A+B)y - \pi_1 - \pi_2 \\ \text{subject to} & x^T B \le \pi_2 \quad \text{and} \quad Ay \le \pi_1 \\ & x, y \in \Delta^n \quad \text{and} \quad \pi_1, \pi_2 \in [-1, 1] \end{array}$$

Nash eq: $e_i^T A y \le x^T A y \quad \forall i$ and

$$x^T B e_j \le x^T B y \quad \forall j$$

Bilinear Program for Nash Eq. [MS'64]

$$\begin{aligned} \text{maximize} \quad & x^T C y - \pi_1 - \pi_2 \\ \text{subject to} \quad & x^T B \le \pi_2 \quad \text{and} \quad A y \le \pi_1 \\ & x, y \in \Delta^n \quad \text{and} \quad \pi_1, \pi_2 \in [-1, 1] \end{aligned}$$

Bilinear Program for Nash Eq. [MS'64]

$$
\begin{aligned}
\text{maximize} \quad & x^T C y - \pi_1 - \pi_2 \\
\text{subject to} \quad & x^T B \le \pi_2 \quad \text{and} \quad Ay \le \pi_1 \\
& x, y \in \Delta^n \quad \text{and} \quad \pi_1, \pi_2 \in [-1, 1]
\end{aligned}
$$

Bilinear Program for Nash Eq. [MS'64]

$$\begin{aligned} \text{maximize} \quad & x^T C y - \pi_1 - \pi_2 \\ \text{subject to} \quad & x^T B \le \pi_2 \quad \text{and} \quad Ay \le \pi_1 \\ & x, y \in \Delta^n \quad \text{and} \quad \pi_1, \pi_2 \in [-1, 1] \end{aligned}$$

Say (x^*, y^*) is a Nash eq. Given $u^* = Cy^*$ we get an LP.

$$\begin{aligned} \text{maximize} \quad & x^T u^* - \pi_1 - \pi_2 \\ \text{subject to} \quad & x^T B \le \pi_2 \quad \text{and} \quad Ay \le \pi_1 \\ & x, y \in \Delta^n \quad \text{and} \quad \pi_1, \pi_2 \in [-1, 1] \\ & Cy = u^* \end{aligned}$$

Bilinear Program for Nash Eq. [MS'64]

$$\begin{aligned} \text{maximize} \quad & x^T C y - \pi_1 - \pi_2 \\ \text{subject to} \quad & x^T B \leq \pi_2 \quad \text{and} \quad Ay \leq \pi_1 \\ & x, y \in \Delta^n \quad \text{and} \quad \pi_1, \pi_2 \in [-1, 1] \end{aligned}$$

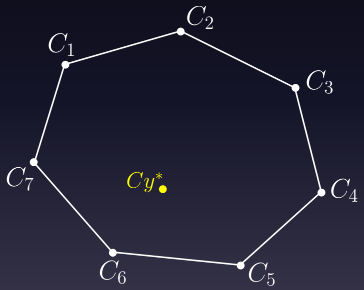
Say (x^*, y^*) is a Nash eq. Given $u^* = Cy^*$ we get an LP.

$$\begin{aligned} \text{maximize} \quad & x^T u^* - \pi_1 - \pi_2 \\ \text{subject to} \quad & x^T B \leq \pi_2 \quad \text{and} \quad Ay \leq \pi_1 \\ & x, y \in \Delta^n \quad \text{and} \quad \pi_1, \pi_2 \in [-1, 1] \\ & Cy = u^* \end{aligned}$$

A vector *close* to Cy^* is sufficient to find an approx. Nash eq.

Bilinear Program for Nash Eq. [MS'64]

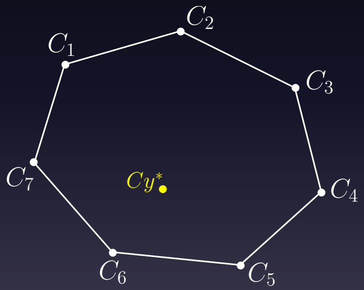
$$\begin{array}{ll} \text{maximize} & x^T C y - \pi_1 - \pi_2 \\ \text{subject to} & x^T B \le \pi_2 \quad \text{and} \quad Ay \le \pi_1 \\ & x, y \in \Delta^n \quad \text{and} \quad \pi_1, \pi_2 \in [-1, 1] \end{array}$$



Matrix C

Bilinear Program for Nash Eq. [MS'64]

$$\begin{aligned} \text{maximize} \quad & x^T C y - \pi_1 - \pi_2 \\ \text{subject to} \quad & x^T B \leq \pi_2 \quad \text{and} \quad Ay \leq \pi_1 \\ & x, y \in \Delta^n \quad \text{and} \quad \pi_1, \pi_2 \in [-1, 1] \end{aligned}$$

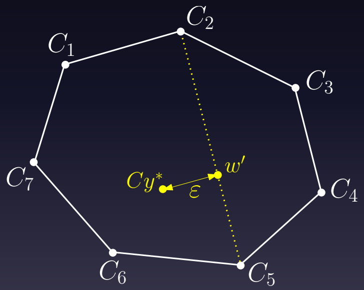


Matrix C

Note: If C_i is s-sparse then $\|C_i\|_{\log s} \leq 2$

Bilinear Program for Nash Eq. [MS'64]

$$\begin{aligned} \text{maximize} \quad & x^T C y - \pi_1 - \pi_2 \\ \text{subject to} \quad & x^T B \le \pi_2 \quad \text{and} \quad Ay \le \pi_1 \\ & x, y \in \Delta^n \quad \text{and} \quad \pi_1, \pi_2 \in [-1, 1] \end{aligned}$$

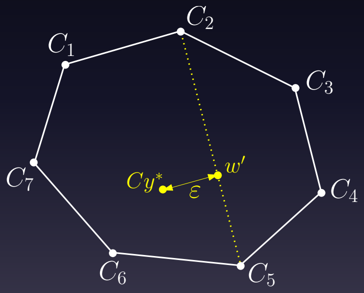


Matrix C

Note: If C_i is s-sparse then $\|C_i\|_{\log s} \le 2$

Bilinear Program for Nash Eq. [MS'64]

$$\begin{aligned} \text{maximize} \quad & x^T C y - \pi_1 - \pi_2 \\ \text{subject to} \quad & x^T B \leq \pi_2 \quad \text{and} \quad Ay \leq \pi_1 \\ & x, y \in \Delta^n \quad \text{and} \quad \pi_1, \pi_2 \in [-1, 1] \end{aligned}$$

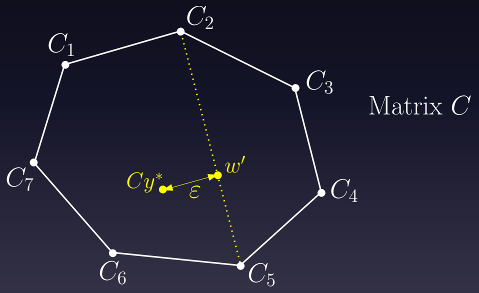


Matrix C

Idea: Exhaustively search for w'

Bilinear Program for Nash Eq. [MS'64]

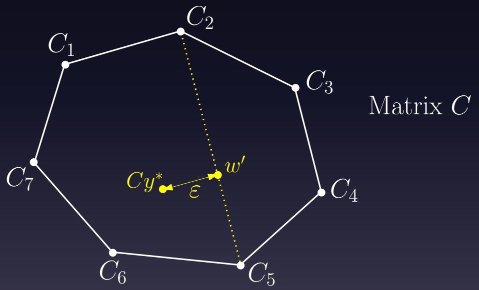
$$\begin{aligned} \text{maximize} \quad & x^T C y - \pi_1 - \pi_2 \\ \text{subject to} \quad & x^T B \leq \pi_2 \quad \text{and} \quad Ay \leq \pi_1 \\ & x, y \in \Delta^n \quad \text{and} \quad \pi_1, \pi_2 \in [-1, 1] \end{aligned}$$



Idea: Exhaustively search for w',
by enumerating subsets of columns of C.

Bilinear Program for Nash Eq. [MS'64]

$$\begin{aligned} \text{maximize} \quad & x^T C y - \pi_1 - \pi_2 \\ \text{subject to} \quad & x^T B \le \pi_2 \quad \text{and} \quad Ay \le \pi_1 \\ & x, y \in \Delta^n \quad \text{and} \quad \pi_1, \pi_2 \in [-1, 1] \end{aligned}$$

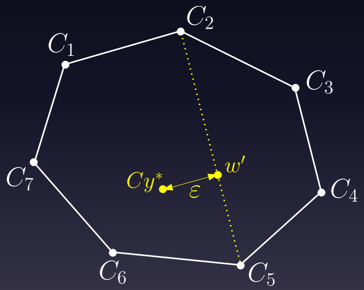


Theorem

In a two-player s-sparse game an ε-Nash equilibrium can be computed in time $n^{O\left(\log s/\varepsilon^2\right)}$.

Bilinear Program for Nash Eq. [MS'64]

$$\begin{aligned} &\text{maximize} && x^T C y - \pi_1 - \pi_2 \\ &\text{subject to} && x^T B \le \pi_2 \quad \text{and} \quad Ay \le \pi_1 \\ & && x, y \in \Delta^n \quad \text{and} \quad \pi_1, \pi_2 \in [-1, 1] \end{aligned}$$



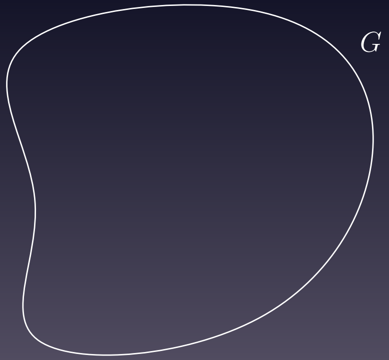
General Result

We can efficiently approximate any sparse bilinear or quadratic form over the simplex.

Application II: Approximation Algorithm for Densest Subgraph

Normalized Densest Subgraph Problem

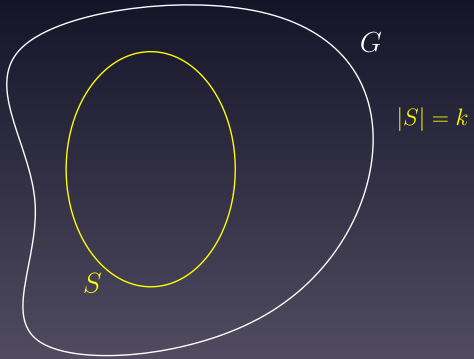
Given: Graph G and size parameter k



Normalized Densest Subgraph Problem

Given: Graph G and size parameter k

Objective: Find vertex subset S of size k such that density(S) is maximized.

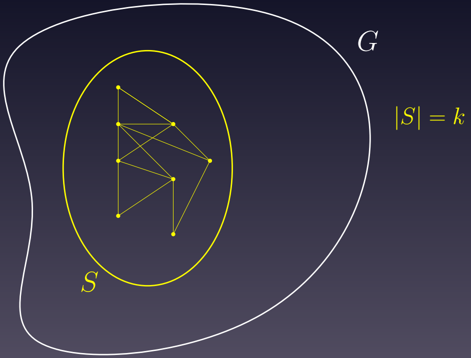


Normalized Densest Subgraph Problem

Given: Graph G and size parameter k

Objective: Find vertex subset S of size k such that density(S) is maximized.

$$\text{density}(S) := \frac{\#\text{ edges in } S}{k^2}$$



Normalized Densest Subgraph Problem

Given: Graph G and size parameter k

Objective: Find vertex subset S of size k such that density(S) is maximized.

$$\text{density}(S) := \frac{\#\ \text{edges in}\ S}{k^2}$$

Theorem

In a degree d graph, an ε additive approximation for the densest bipartite subgraph problem can be computed in time

$$n^{O\left(\varepsilon^{-2}\log(d/k)\right)}.$$

✓ Application I: Approximating Nash Equilibria

✓ Application II: Approximating Dense Subgraphs

General Result

We can efficiently approximate any sparse bilinear or quadratic form over the simplex.

Extensions

- Convex hull of matrices with entrywise norm and Schatten p-norm
- Shapley-Folkman Lemma
- Colorful Carathéodory Theorem
- Finding close vectors via linear optimization oracles (Mirrokni et al., 2015)

Extensions

- Convex hull of matrices with entrywise norm and Schatten p-norm
- Shapley-Folkman Lemma
- Colorful Carathéodory Theorem
- Finding close vectors via linear optimization oracles (Mirrokni et al., 2015)

Thank You!

Khintchine Inequality

Let $r_1, r_2, \ldots, r_m$ be a sequence of i.i.d. random variables with $\Pr(r_i = \pm 1) = \frac{1}{2}$
In addition, let $u_1, u_2, \ldots, u_m \in \mathbb{R}^d$ be a deterministic sequence of vectors. Then, for $2 \leq p < \infty$

$$\mathbb{E} \left\| \sum_{i=1}^{m} r_i u_i \right\|_p \leq \sqrt{p} \; \left(\sum_{i=1}^{m} \|u_i\|_p^2 \right)^{\frac{1}{2}}$$