An Information Complexity Approach to Extended Formulations

Ankur Moitra
Institute for Advanced Study

joint work with Mark Braverman

The Permutahedron

The Permutahedron

Let $\vec{t} = [1, 2, 3, \ldots n]$, $P = \text{conv}\left\{\pi(\vec{t}) \,\middle|\, \pi \text{ is permutation}\right\}$

The Permutahedron

Let $\vec{t} = [1, 2, 3, \ldots n]$, $P = \text{conv}\{\pi(\vec{t}) \mid \pi \text{ is permutation}\}$

How many facets of P have?

The Permutahedron

Let $\vec{t} = [1, 2, 3, \ldots n]$, $P = \text{conv}\{\pi(\vec{t}) \mid \pi \text{ is permutation}\}$

How many facets of P have?

exponentially many!

The Permutahedron

Let $\vec{t} = [1, 2, 3, \ldots n]$, $P = \mathrm{conv}\{\pi(\vec{t}) \mid \pi \text{ is permutation}\}$

How many facets of P have?

exponentially many!

e.g. $S \subset [n]$, $\Sigma_{i \text{ in } S}\, x_i \geq 1 + 2 + \ldots + |S| = |S|(|S|+1)/2$

The Permutahedron

Let $\vec{t} = [1, 2, 3, \ldots n]$, $P = \text{conv}\{\pi(\vec{t}) \mid \pi \text{ is permutation}\}$

How many facets of P have?

exponentially many!

e.g. $S \subset [n]$, $\Sigma_{i \text{ in } S}\, x_i \geq 1 + 2 + \ldots + |S| = |S|(|S|+1)/2$

Let $Q = \{A \mid A \text{ is doubly-stochastic}\}$

The Permutahedron

Let $\vec{t} = [1, 2, 3, \ldots n]$, $P = \text{conv}\{\pi(\vec{t}) \mid \pi \text{ is permutation}\}$

How many facets of P have?

exponentially many!

e.g. $S \subset [n]$, $\Sigma_{i \text{ in } S}\, x_i \geq 1 + 2 + \ldots + |S| = |S|(|S|+1)/2$

Let $Q = \{A \mid A \text{ is doubly-stochastic}\}$

Then P is the projection of Q: $P = \{A\vec{t} \mid A \text{ in } Q\}$

The Permutahedron

Let $\vec{t} = [1, 2, 3, \ldots n]$, $P = \text{conv}\{\pi(\vec{t}) \mid \pi \text{ is permutation}\}$

How many facets of P have?

exponentially many!

e.g. $S \subset [n]$, $\Sigma_{i \text{ in } S}\, x_i \geq 1 + 2 + \ldots + |S| = |S|(|S|+1)/2$

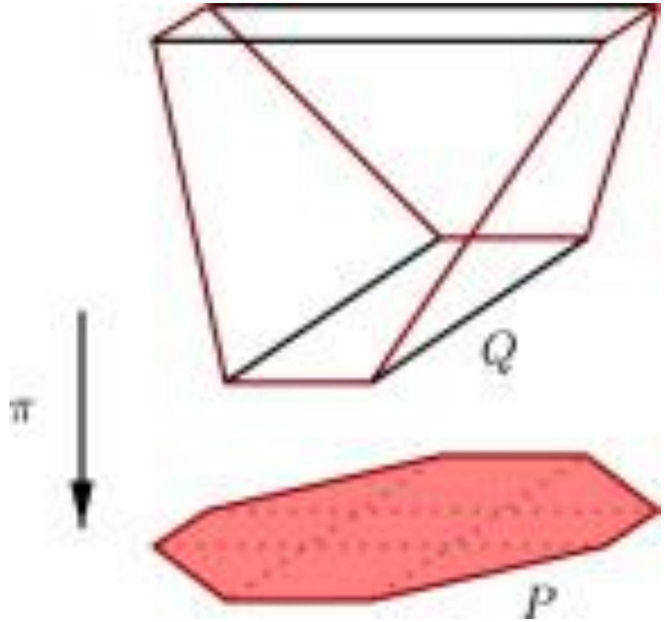
Let $Q = \{A \mid A \text{ is doubly-stochastic}\}$

Then P is the projection of Q: $P = \{A\vec{t} \mid A \text{ in } Q\}$

Yet Q has only $O(n^2)$ facets

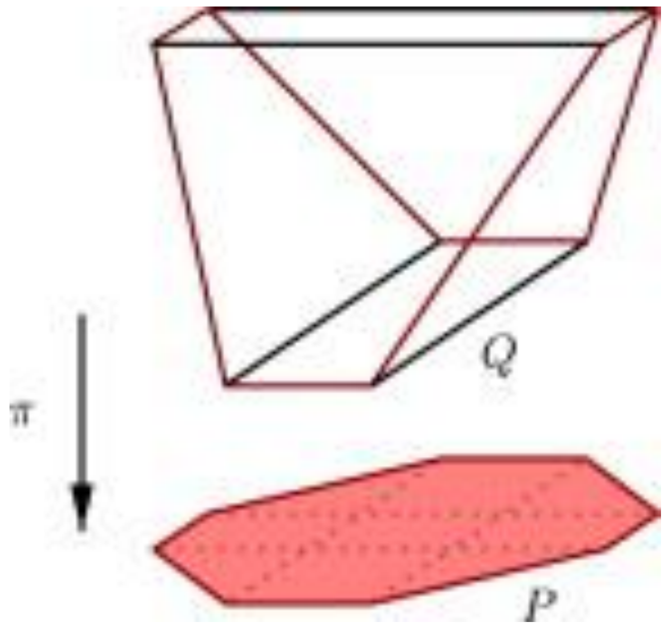
Extended Formulations

The **extension complexity (xc)** of a polytope P is the minimum number of facets of Q so that P = proj(Q)



Extended Formulations

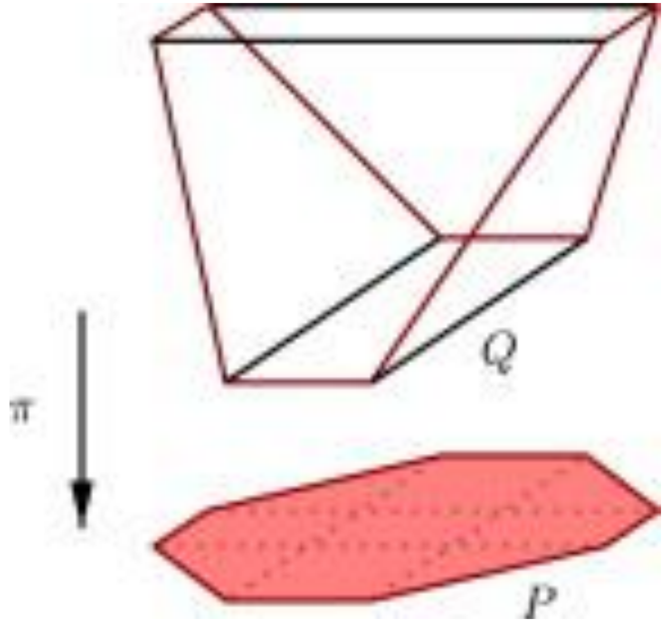
The **extension complexity (xc)** of a polytope P is the minimum number of facets of Q so that P = proj(Q)



e.g. xc(P) = Θ(n logn)
for permutahedron

Extended Formulations

The **extension complexity (xc)** of a polytope P is the minimum number of facets of Q so that P = proj(Q)

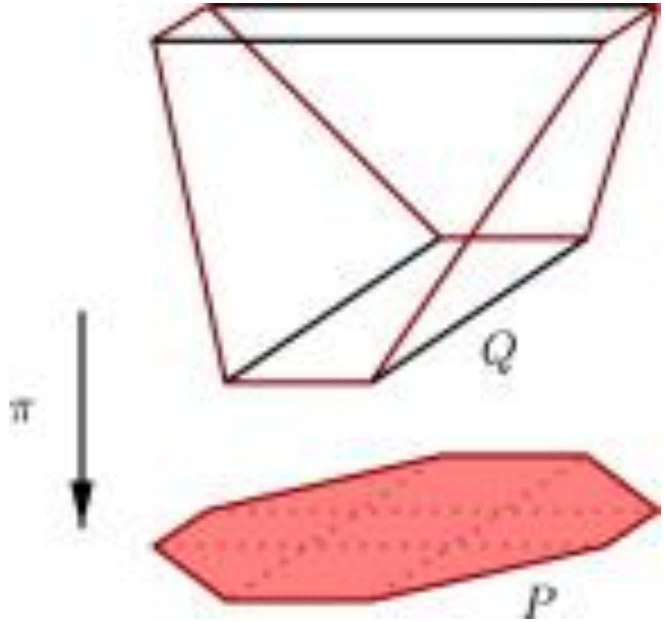


e.g. xc(P) = $\Theta(n \log n)$
for permutahedron

xc(P) = $\Theta(\log n)$ for a regular n-gon, but $\Omega(\sqrt{n})$ for its perturbation

Extended Formulations

The **extension complexity (xc)** of a polytope P is the minimum number of facets of Q so that P = proj(Q)



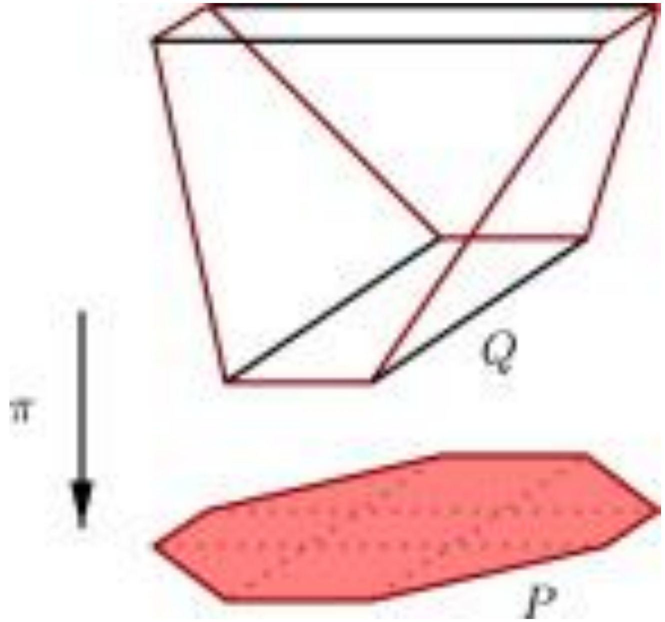
e.g. xc(P) = Θ(n logn)
for permutahedron

xc(P) = Θ(logn) for a regular n-gon, but Ω(√n) for its perturbation

In general, P = {x | ∃ y, (x,y) in Q}

Extended Formulations

The **extension complexity (xc)** of a polytope P is the minimum number of facets of Q so that P = proj(Q)



e.g. xc(P) = Θ(n logn)
for permutahedron

xc(P) = Θ(logn) for a regular n-gon, but Ω(√n) for its perturbation

In general, P = {x | ∃ y, (x,y) in Q}

…analogy with **quantifiers** in Boolean formulae

Applications of EFs

In general, $P = \{x \mid \exists y, (x,y) \text{ in } Q\}$

Applications of EFs

In general, $P = \{x \mid \exists y, (x,y) \text{ in } Q\}$

Through EFs, we can reduce # facets exponentially!

Applications of EFs

In general, $P = \{x \mid \exists y, (x,y) \text{ in } Q\}$

Through EFs, we can reduce # facets exponentially!

Hence, we can run standard LP solvers instead of the ellipsoid algorithm

Applications of EFs

In general, $P = \{x \mid \exists\, y, (x,y) \text{ in } Q\}$

Through EFs, we can reduce # facets exponentially!

Hence, we can run standard LP solvers instead of the ellipsoid algorithm

EFs often give, or are based on new combinatorial insights

Applications of EFs

In general, $P = \{x \mid \exists y, (x,y) \text{ in } Q\}$

Through EFs, we can reduce # facets exponentially!

Hence, we can run standard LP solvers instead of the ellipsoid algorithm

EFs often give, or are based on new combinatorial insights

e.g. Birkhoff-von Neumann Thm and permutahedron

Applications of EFs

In general, $P = \{x \mid \exists y, (x,y) \text{ in } Q\}$

Through EFs, we can reduce # facets exponentially!

Hence, we can run standard LP solvers instead of the ellipsoid algorithm

EFs often give, or are based on new combinatorial insights

e.g. Birkhoff-von Neumann Thm and permutahedron

e.g. prove there is low-cost object, through its polytope

Explicit, Hard Polytopes?

Explicit, Hard Polytopes?

Definition: TSP polytope:

$$P = \text{conv}\{\mathbf{1}_F \mid F \text{ is the set of edges on a tour of } K_n\}$$

Explicit, Hard Polytopes?

Definition: TSP polytope:

$$P = \mathrm{conv}\{\mathbf{1}_F \mid F \text{ is the set of edges on a tour of } K_n\}$$

(If we could optimize over this polytope, then P = NP)

Explicit, Hard Polytopes?

Definition: TSP polytope:

$$P = \text{conv}\{\mathbf{1}_F \mid F \text{ is the set of edges on a tour of } K_n\}$$

(If we could optimize over this polytope, then P = NP)

Can we prove **unconditionally** there is no small EF?

Explicit, Hard Polytopes?

Definition: TSP polytope:

$$P = \text{conv}\{\mathbf{1}_F \mid F \text{ is the set of edges on a tour of } K_n\}$$

(If we could optimize over this polytope, then P = NP)

Can we prove **unconditionally** there is no small EF?

Caveat: this is unrelated to proving complexity l.b.s

Explicit, Hard Polytopes?

Definition: TSP polytope:

$$P = \text{conv}\{\mathbf{1}_F \mid F \text{ is the set of edges on a tour of } K_n\}$$

(If we could optimize over this polytope, then P = NP)

Can we prove **unconditionally** there is no small EF?

Caveat: this is unrelated to proving complexity l.b.s

[Yannakakis ’90]: Yes, through the **nonnegative rank**

Theorem [Yannakakis '90]: Any symmetric LP for TSP or matching has size $2^{\Omega(n)}$

Theorem [Yannakakis '90]: Any symmetric LP for TSP or matching has size $2^{\Omega(n)}$

Theorem [Fiorini et al '12]: Any LP for TSP has size $2^{\Omega(\sqrt{n})}$ (based on a $2^{\Omega(n)}$ lower bd for clique)

Theorem [Yannakakis ’90]: Any symmetric LP for TSP or matching has size $2^{\Omega(n)}$

Theorem [Fiorini et al ’12]: Any LP for TSP has size $2^{\Omega(\sqrt{n})}$ (based on a $2^{\Omega(n)}$ lower bd for clique)

Theorem [Braun et al ’12]: Any LP that approximates clique within $n^{1/2\text{-eps}}$ has size $\exp(n^{\text{eps}})$

Theorem [Yannakakis '90]: Any symmetric LP for TSP or matching has size $2^{\Omega(n)}$

Theorem [Fiorini et al '12]: Any LP for TSP has size $2^{\Omega(\sqrt{n})}$ (based on a $2^{\Omega(n)}$ lower bd for clique)

Theorem [Braun et al '12]: Any LP that approximates clique within $n^{1/2-eps}$ has size $\exp(n^{eps})$

Hastad's proved an $n^{1-o(1)}$ hardness of approx. for clique, can we prove the analogue for EFs?

Theorem [Yannakakis '90]: Any symmetric LP for TSP or matching has size $2^{\Omega(n)}$

Theorem [Fiorini et al '12]: Any LP for TSP has size $2^{\Omega(\sqrt{n})}$ (based on a $2^{\Omega(n)}$ lower bd for clique)

Theorem [Braun et al '12]: Any LP that approximates clique within $n^{1/2-eps}$ has size $\exp(n^{eps})$

Hastad's proved an $n^{1-o(1)}$ hardness of approx. for clique, can we prove the analogue for EFs?

Theorem [Braverman, Moitra '13]: Any LP that approximates clique within n^{1-eps} has size $\exp(n^{eps})$

Outline

Part I: Tools for Extended Formulations

- Yannakakis's Factorization Theorem
- The Rectangle Bound
- A Sampling Argument

Part II: Applications

- Correlation Polytope
- Approximating the Correlation Polytope
- A Better Lower Bound for Disjointness

Outline

Part I: Tools for Extended Formulations

- **Yannakakis's Factorization Theorem**
- The Rectangle Bound
- A Sampling Argument

Part II: Applications

- Correlation Polytope
- Approximating the Correlation Polytope
- A Better Lower Bound for Disjointness

The Factorization Theorem

The Factorization Theorem

How can we prove lower bounds on EFs?

The Factorization Theorem

How can we prove lower bounds on EFs?

[Yannakakis ’90]:

Geometric Parameter $\Longleftrightarrow$ Algebraic Parameter

The Factorization Theorem

How can we prove lower bounds on EFs?

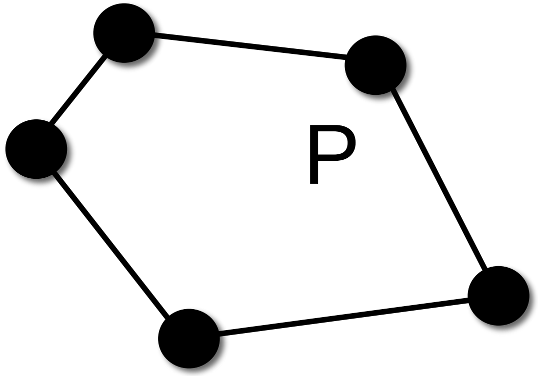
[Yannakakis '90]:

Geometric Parameter ⟺ Algebraic Parameter

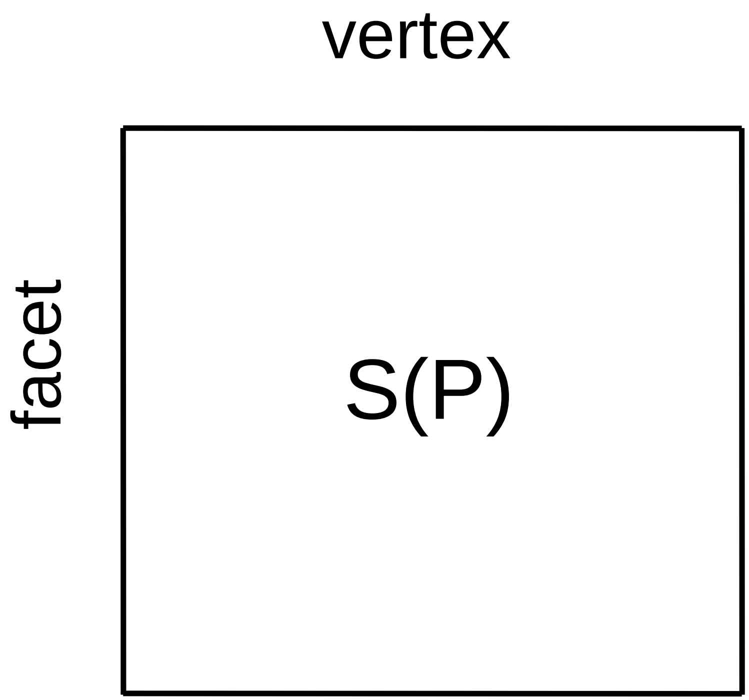
Definition of the **slack matrix**…

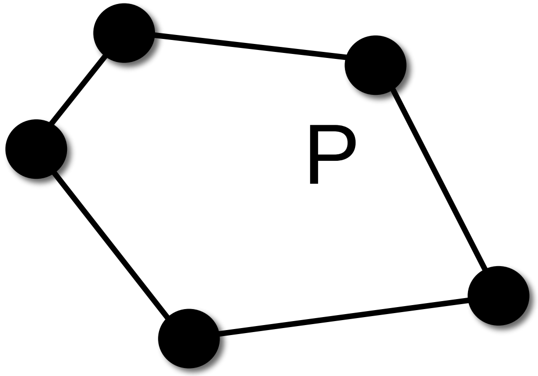
The Slack Matrix

The Slack Matrix

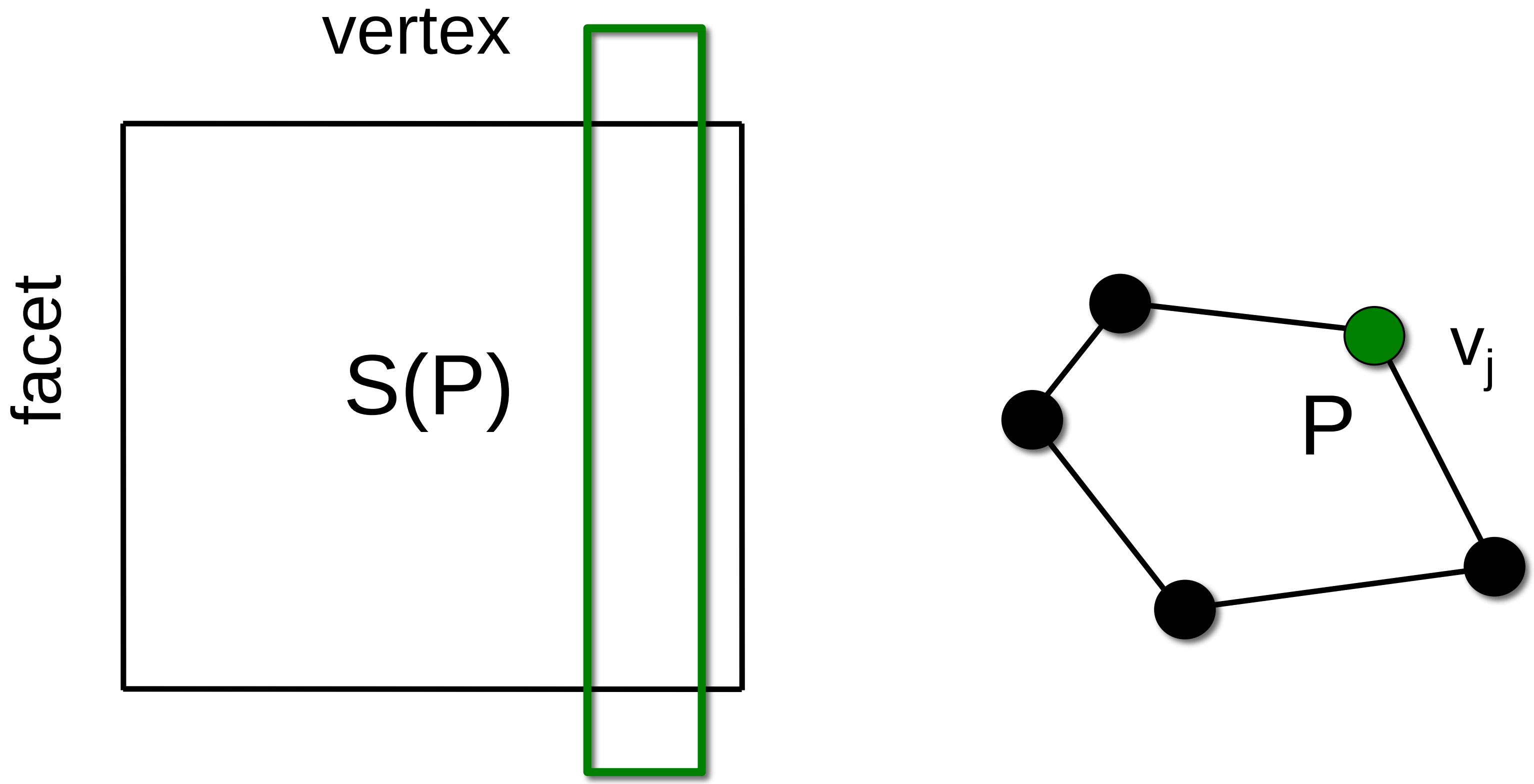


The Slack Matrix

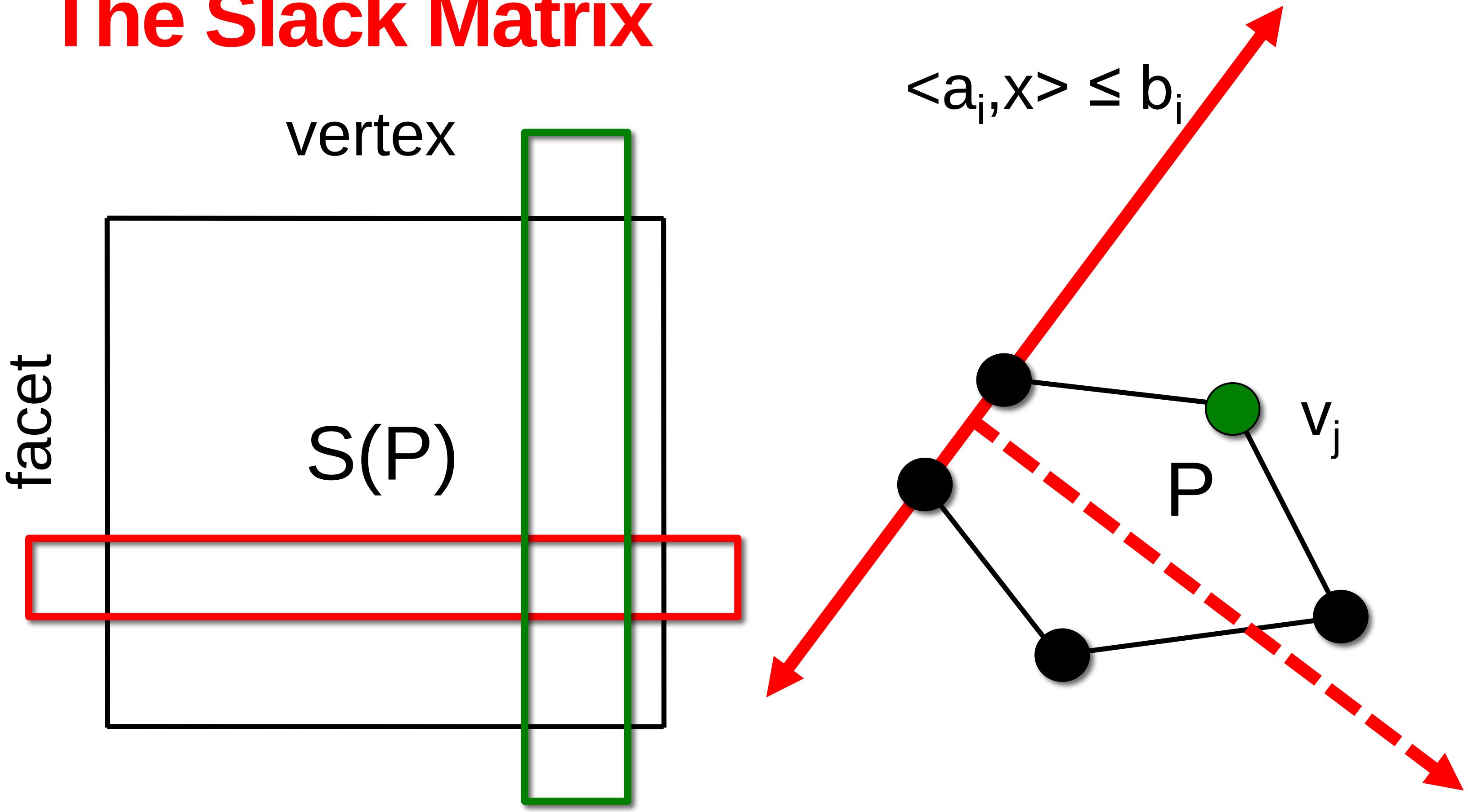




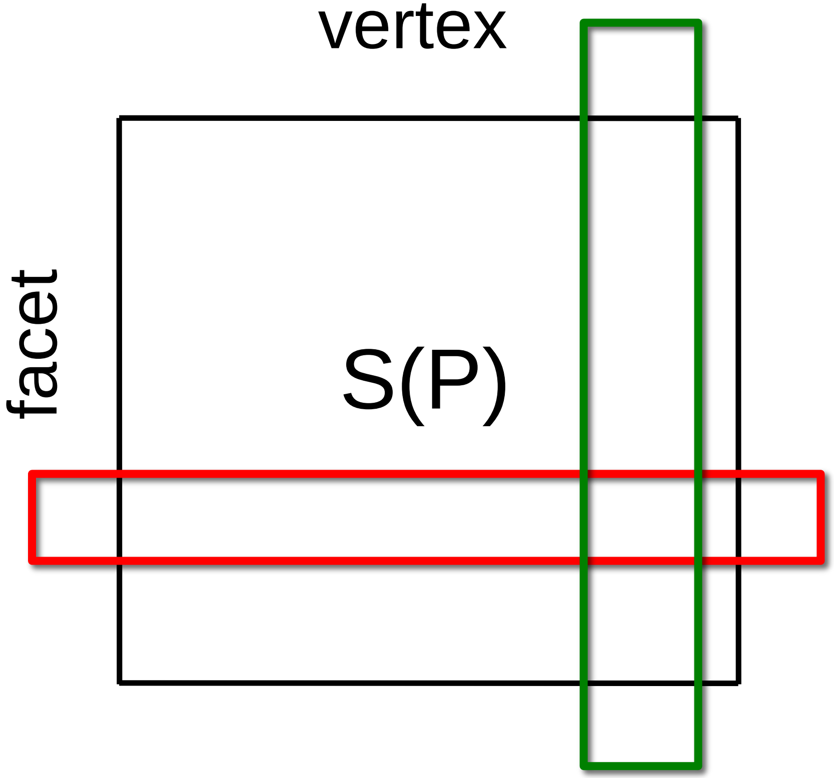
The Slack Matrix

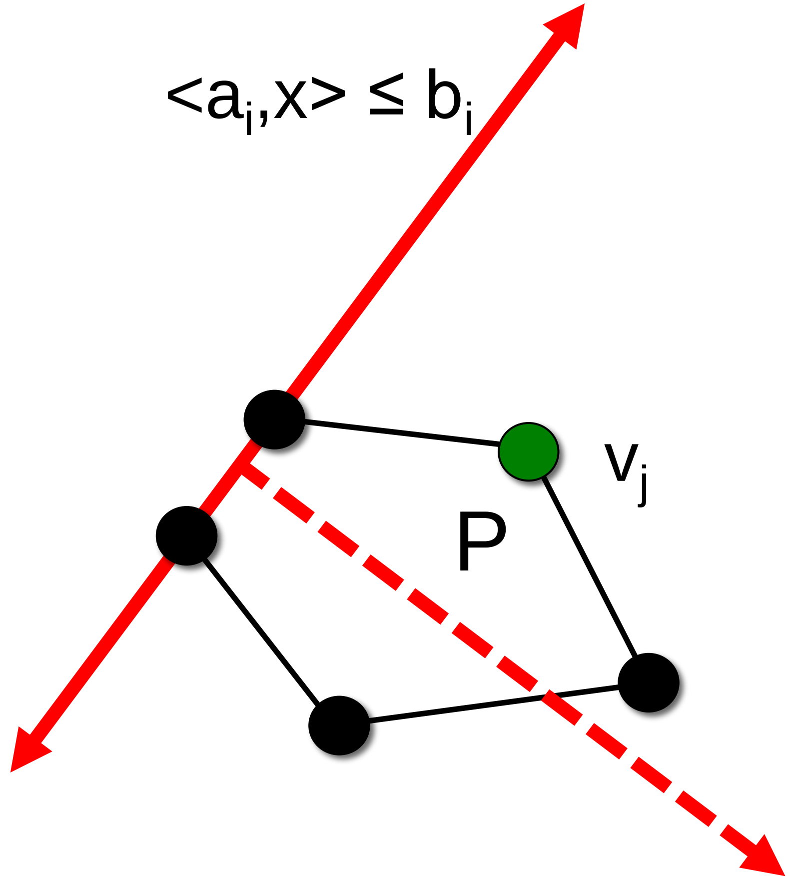


The Slack Matrix



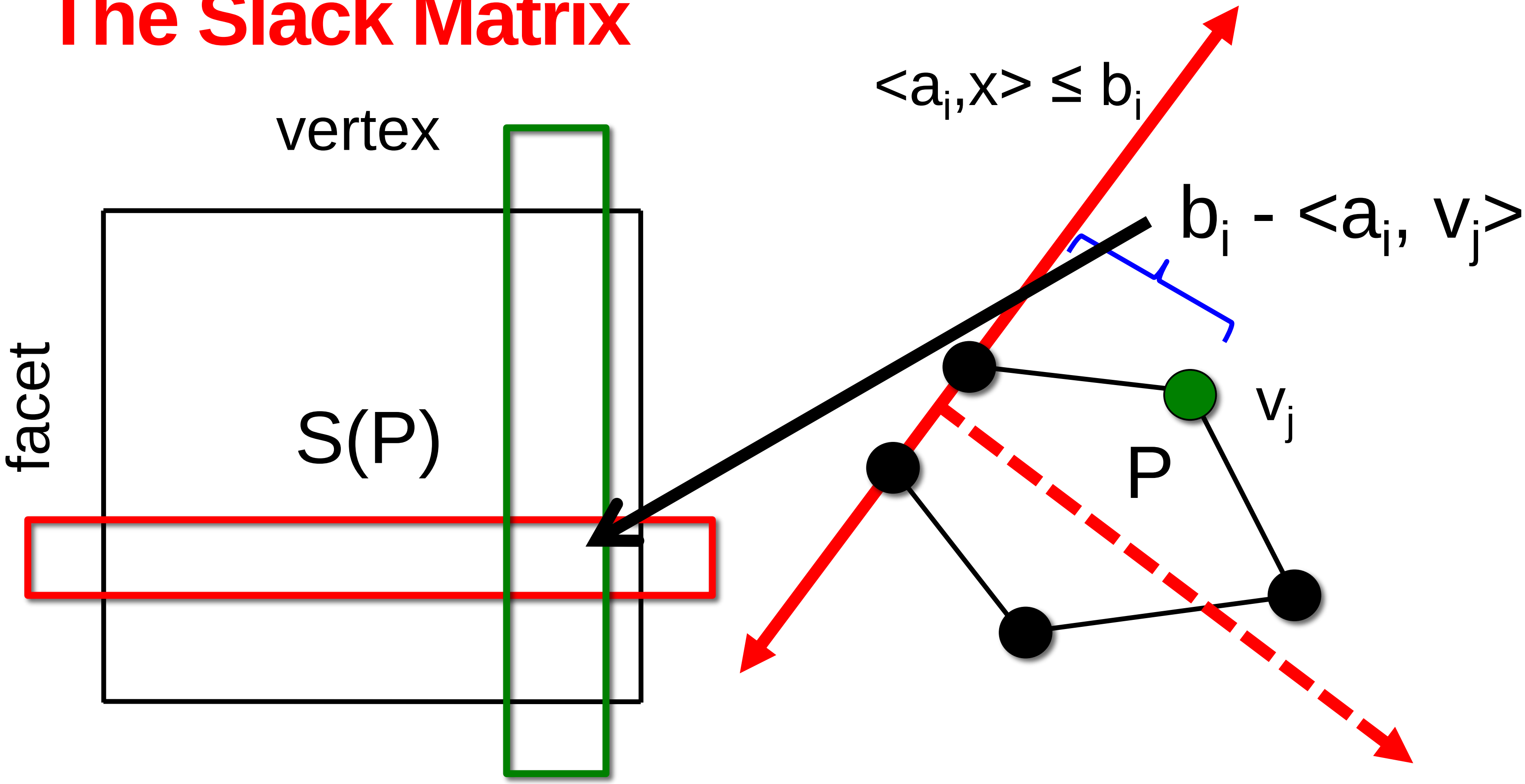
The Slack Matrix





The entry in row i, column j is how *slack* the j^{th} vertex is on the i^{th} constraint

The Slack Matrix



The entry in row i, column j is how *slack* the j^{th} vertex is on the i^{th} constraint

The Factorization Theorem

How can we prove lower bounds on EFs?

[Yannakakis '90]:

Geometric Parameter $\Longleftrightarrow$ Algebraic Parameter

Definition of the **slack matrix**…

The Factorization Theorem

How can we prove lower bounds on EFs?

[Yannakakis ’90]:

Geometric Parameter ⟷ Algebraic Parameter

Definition of the **slack matrix**…

Definition of the **nonnegative rank**…

Nonnegative Rank

$$\boxed{S} =$$

Nonnegative Rank

$$\boxed{S} = \overbrace{\boxed{M_1}}^{\text{rank one, nonnegative}} + \dots + \boxed{M_r}$$

Nonnegative Rank

$$\boxed{S} = \overbrace{\boxed{M_1}}^{\text{rank one, nonnegative}} + \ldots + \boxed{M_r}$$

Definition: $\text{rank}^+(S)$ is the smallest r s.t. S can be written as the sum of r rank one, nonneg. matrices

Nonnegative Rank

$$\boxed{S} = \overbrace{\boxed{M_1}}^{\text{rank one, nonnegative}} + \dots + \boxed{M_r}$$

Definition: $\text{rank}^+(S)$ is the smallest r s.t. S can be written as the sum of r rank one, nonneg. matrices

Note: $\text{rank}^+(S) \geq \text{rank}(S)$, but can be much larger too!

The Factorization Theorem

How can we prove lower bounds on EFs?

[Yannakakis '90]:

Geometric Parameter ⟺ Algebraic Parameter

The Factorization Theorem

How can we prove lower bounds on EFs?

[Yannakakis '90]: $\mathbf{xc(P) = rank^+(S(P))}$

Geometric Parameter ⟺ Algebraic Parameter

The Factorization Theorem

How can we prove lower bounds on EFs?

[Yannakakis '90]: $\mathbf{xc(P) = rank^+(S(P))}$

Geometric Parameter ⟺ Algebraic Parameter

Intuition: the factorization gives a change of variables that preserves the slack matrix!

The Factorization Theorem

How can we prove lower bounds on EFs?

[Yannakakis '90]: $\mathbf{xc(P) = rank^{+}(S(P))}$

Geometric Parameter

Algebraic Parameter

Intuition: the factorization gives a change of variables that preserves the slack matrix!

We will give a new way to lower bound nonnegative rank via **information theory**…

Outline

Part I: Tools for Extended Formulations

- Yannakakis's Factorization Theorem
- The Rectangle Bound
- A Sampling Argument

Part II: Applications

- Correlation Polytope
- Approximating the Correlation Polytope
- A Better Lower Bound for Disjointness

Outline

Part I: Tools for Extended Formulations

- Yannakakis's Factorization Theorem
- **The Rectangle Bound**
- A Sampling Argument

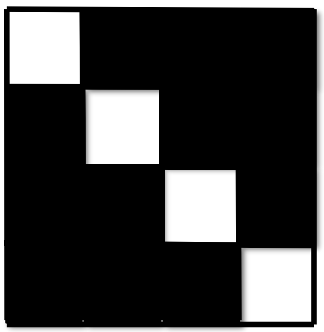
Part II: Applications

- Correlation Polytope
- Approximating the Correlation Polytope
- A Better Lower Bound for Disjointness

The Rectangle Bound

$$S = \overbrace{M_1}^{\text{rank one, nonnegative}} + \dots + M_r$$

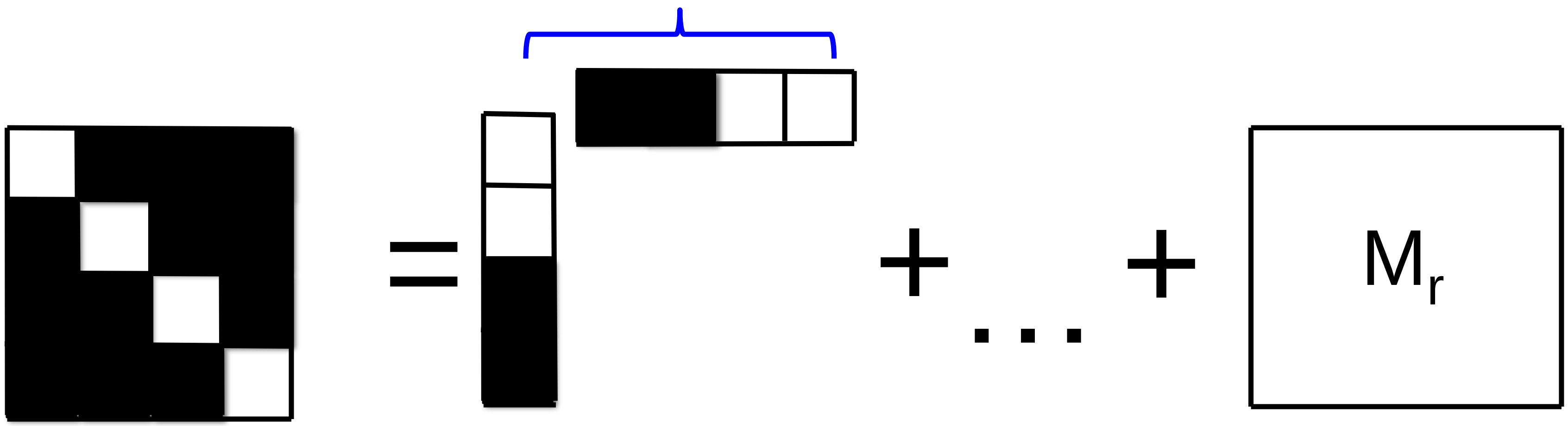
The Rectangle Bound



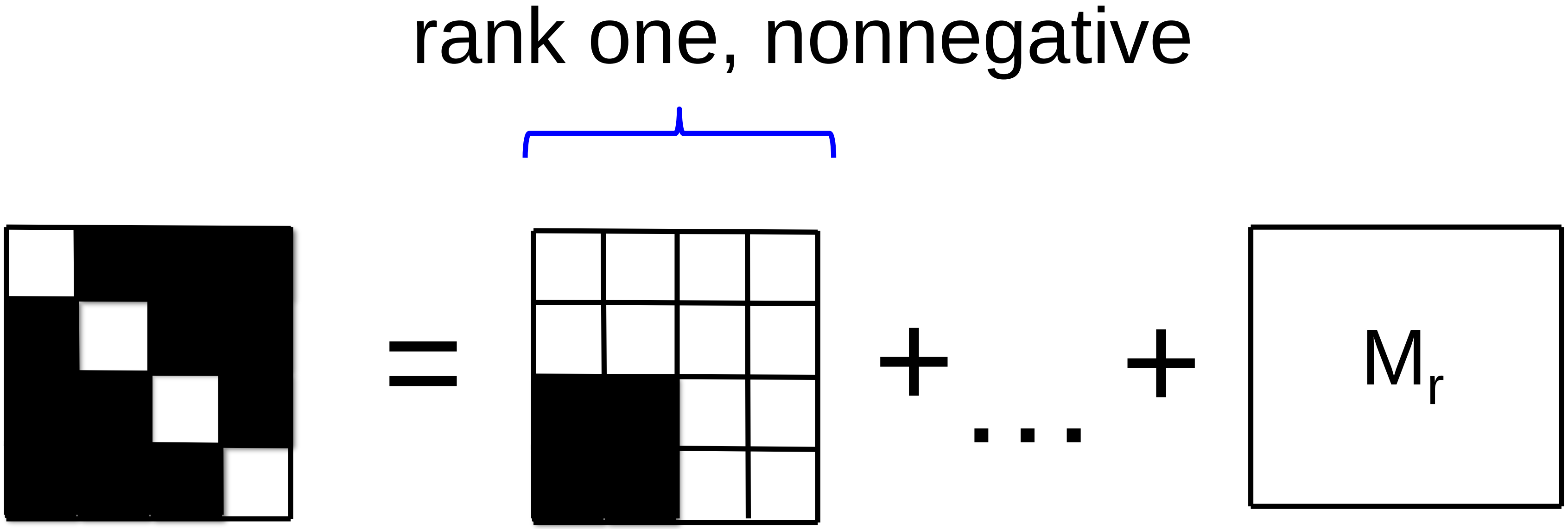
$$= \overbrace{\boxed{M_1}}^{\text{rank one, nonnegative}} + \ldots + \boxed{M_r}$$

The Rectangle Bound

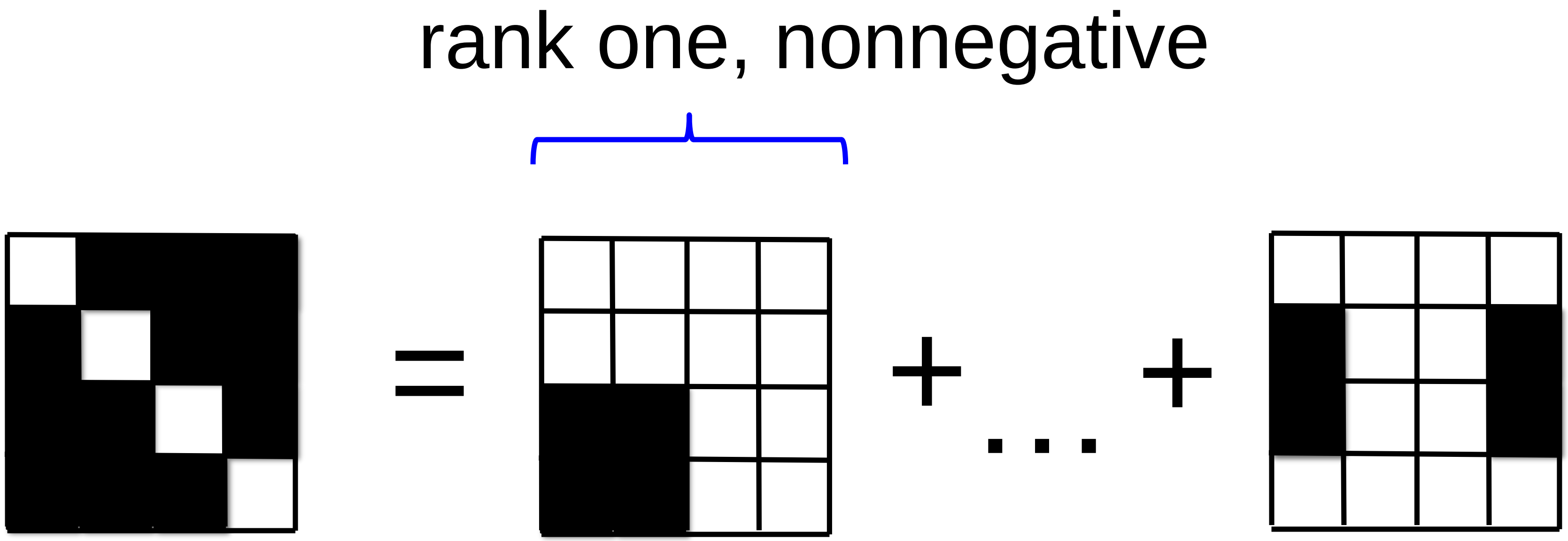
rank one, nonnegative



The Rectangle Bound

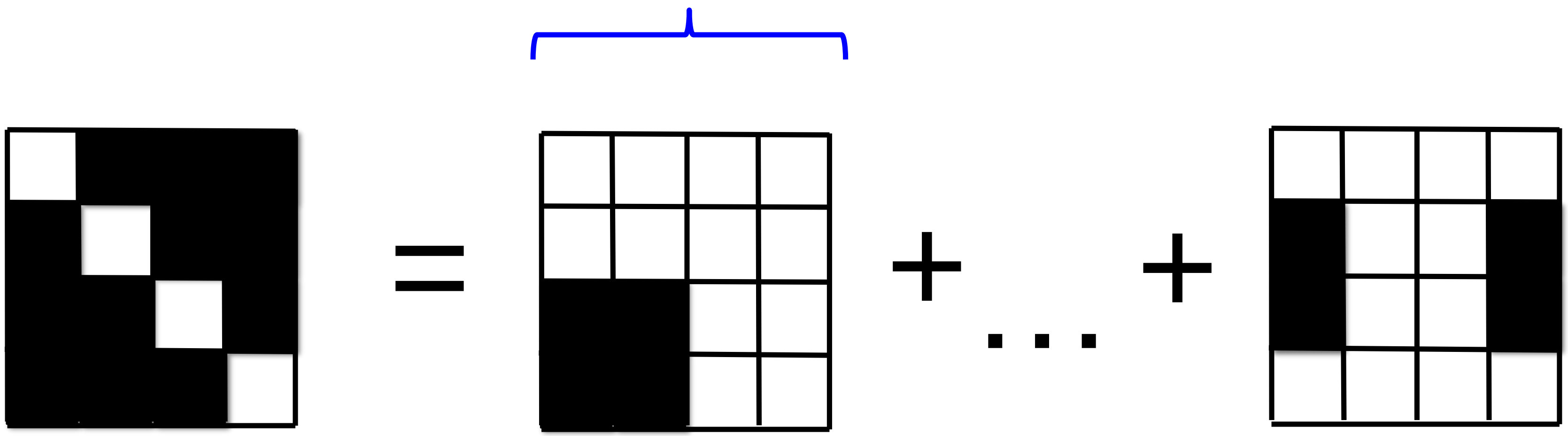


The Rectangle Bound



The Rectangle Bound

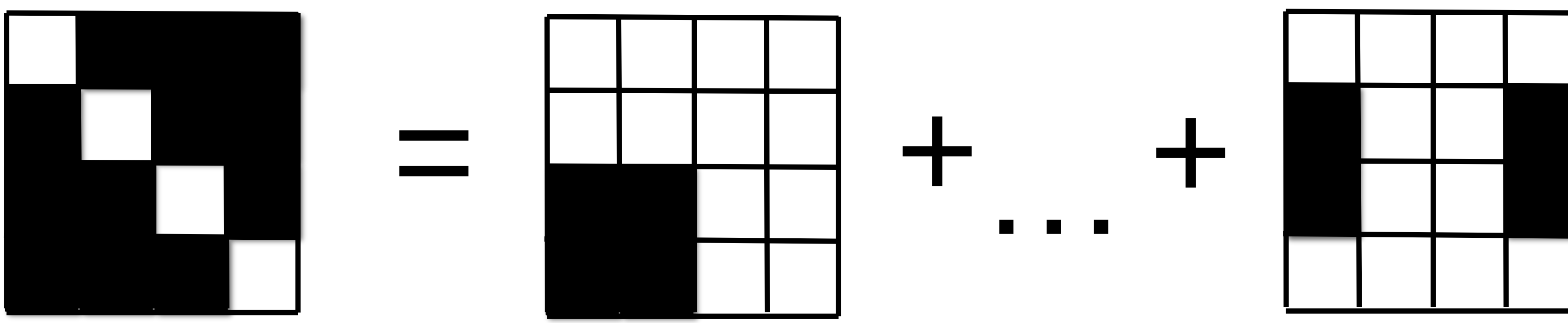
rank one, nonnegative



The support of each M_i is a combinatorial rectangle

The Rectangle Bound

rank one, nonnegative

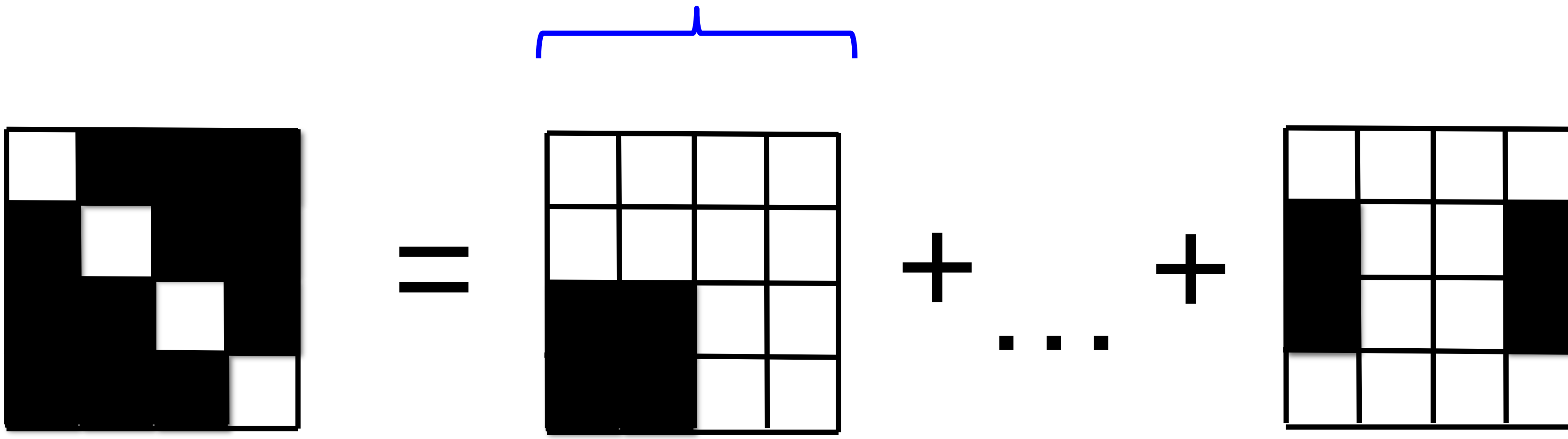


The support of each M_i is a combinatorial rectangle

rank$^+$(S) is at least # rectangles needed to cover supp of S

The Rectangle Bound

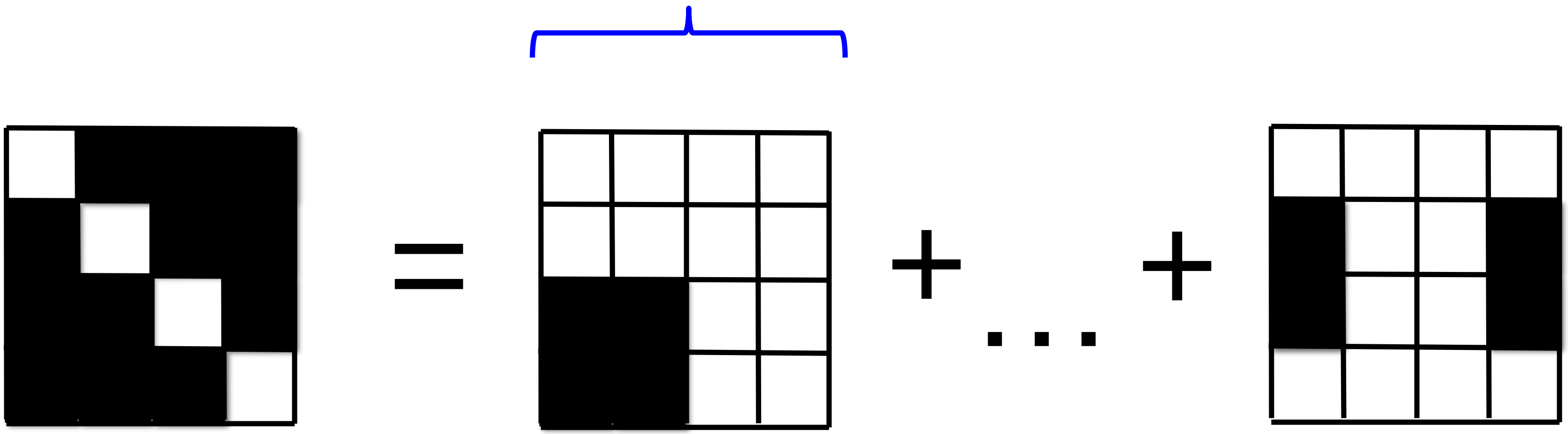
rank one, nonnegative



rank$^+$(S) is at least # rectangles needed to cover supp of S

The Rectangle Bound

rank one, nonnegative



Non-deterministic Comm. Complexity

rank^{+}(S) is at least # rectangles needed to cover supp of S

Outline

Part I: Tools for Extended Formulations

- Yannakakis's Factorization Theorem
- The Rectangle Bound
- A Sampling Argument

Part II: Applications

- Correlation Polytope
- Approximating the Correlation Polytope
- A Better Lower Bound for Disjointness

Outline

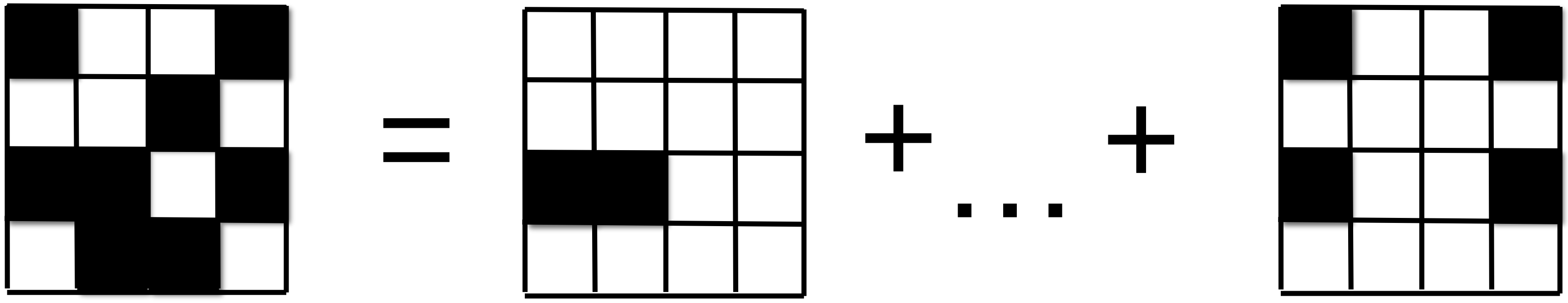
Part I: Tools for Extended Formulations

- Yannakakis's Factorization Theorem
- The Rectangle Bound
- **A Sampling Argument**

Part II: Applications

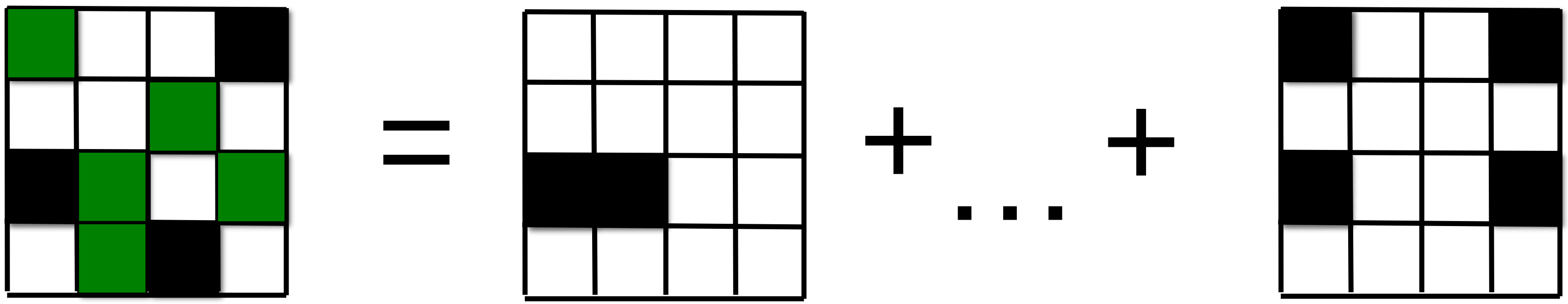
- Correlation Polytope
- Approximating the Correlation Polytope
- A Better Lower Bound for Disjointness

A Sampling Argument



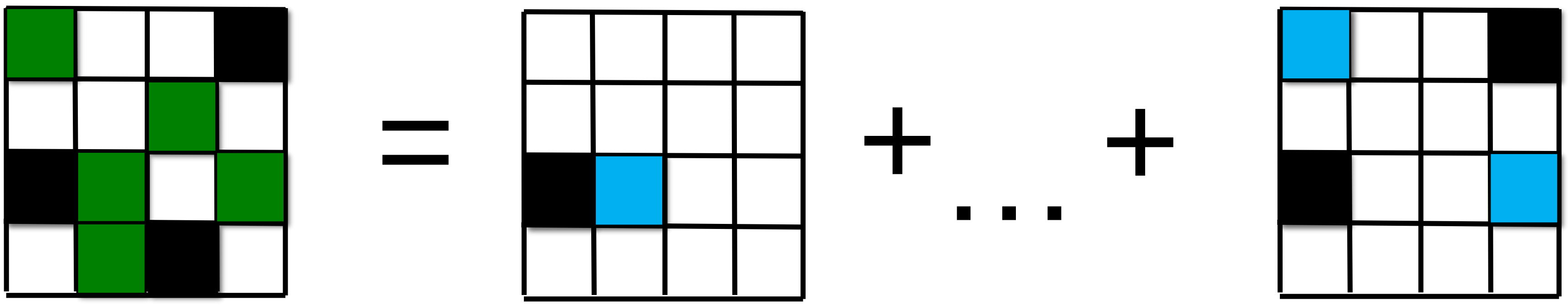
A Sampling Argument

T = { }, set of entries in S with same value



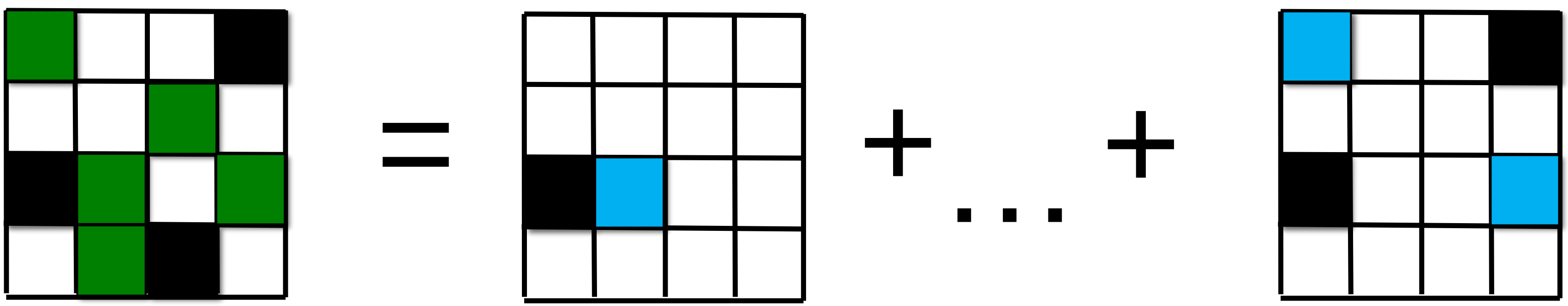
A Sampling Argument

T = {}, set of entries in S with same value



A Sampling Argument

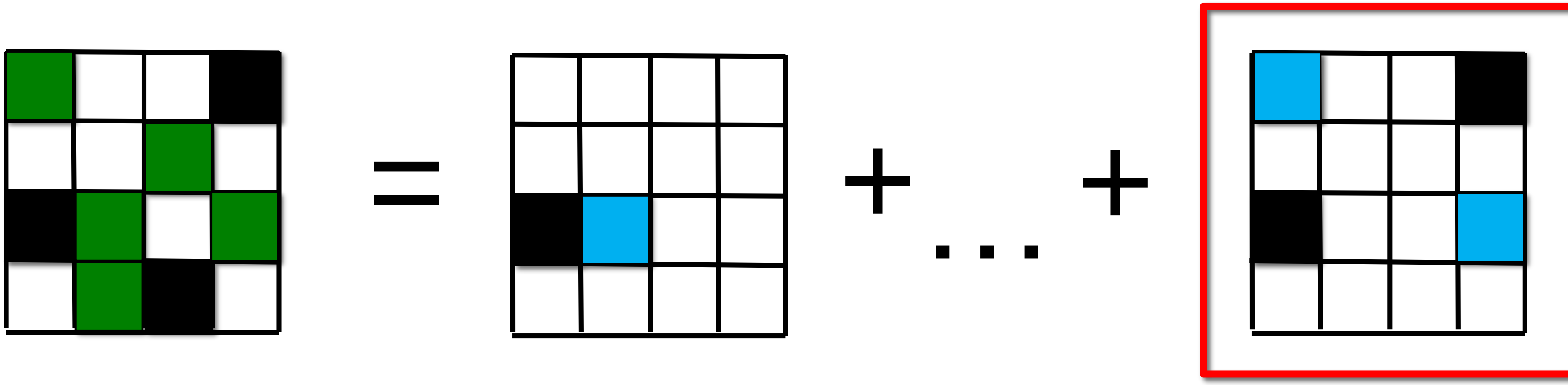
T = {■}, set of entries in S with same value



Choose M_i proportional to total value on T

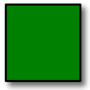
A Sampling Argument

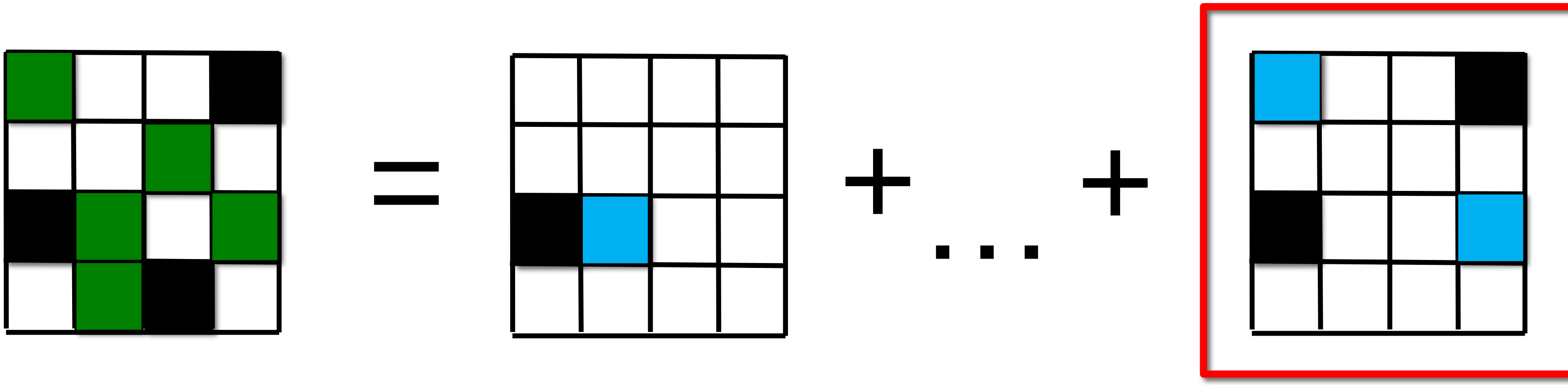
T = {■}, set of entries in S with same value



Choose M_i proportional to total value on T

A Sampling Argument

T = {}, set of entries in S with same value

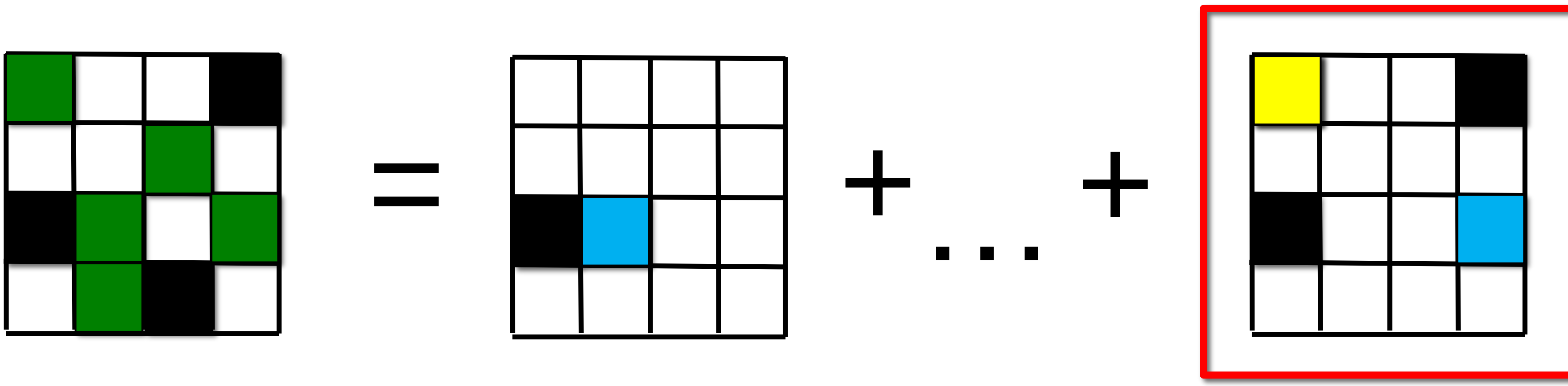


Choose M_i proportional to total value on T

Choose (a,b) in T proportional to relative value in M_i

A Sampling Argument

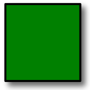
T = { }, set of entries in S with same value

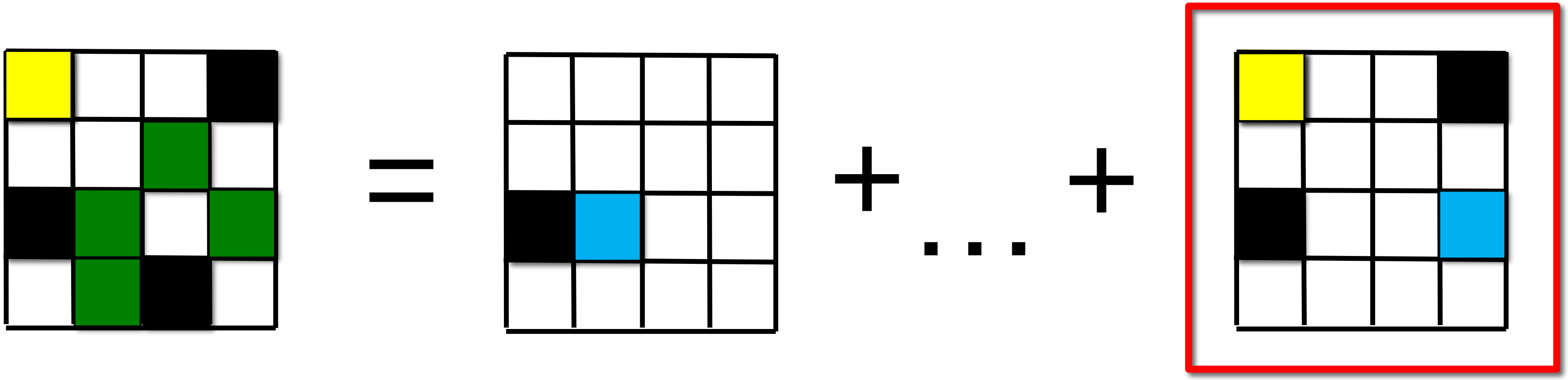


Choose M_i proportional to total value on T

Choose (a,b) in T proportional to relative value in M_i

A Sampling Argument

T = {}, set of entries in S with same value

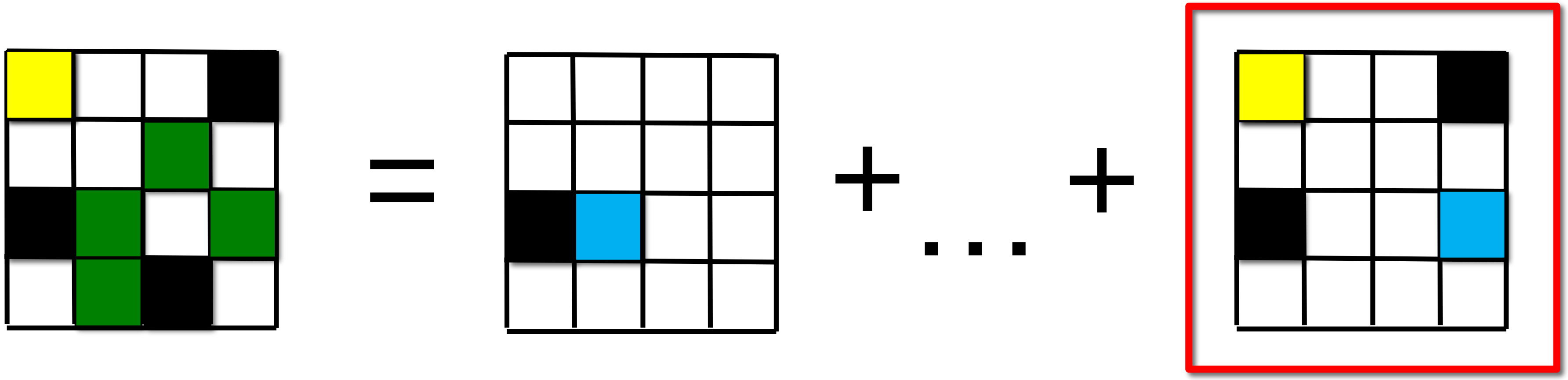


Choose M_i proportional to total value on T

Choose (a,b) in T proportional to relative value in M_i

A Sampling Argument

T = {■}, set of entries in S with same value



Choose M_i proportional to total value on T

Choose (a,b) in T proportional to relative value in M_i

If r is too small, this procedure uses too little entropy!

Outline

Part I: Tools for Extended Formulations

- Yannakakis's Factorization Theorem
- The Rectangle Bound
- A Sampling Argument

Part II: Applications

- Correlation Polytope
- Approximating the Correlation Polytope
- A Better Lower Bound for Disjointness

Outline

Part I: Tools for Extended Formulations

- Yannakakis's Factorization Theorem
- The Rectangle Bound
- A Sampling Argument

Part II: Applications

- **Correlation Polytope**
- Approximating the Correlation Polytope
- A Better Lower Bound for Disjointness

The Construction of [Fiorini et al]

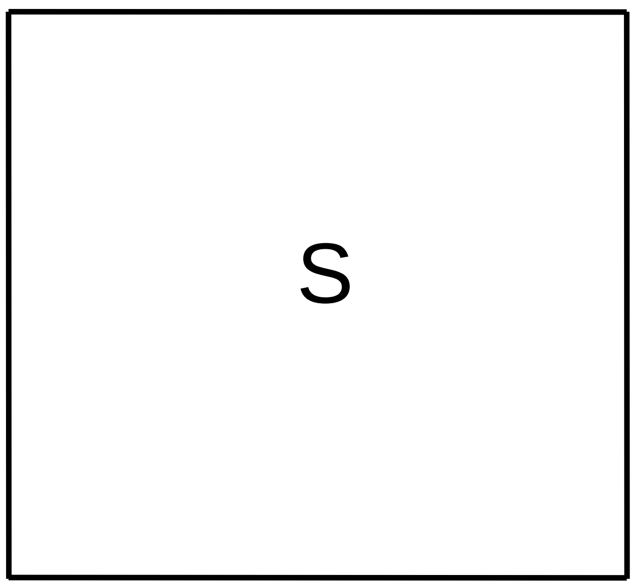
correlation polytope: $P_{corr} = \text{conv}\{aa^T | a \text{ in } \{0,1\}^n\}$

The Construction of [Fiorini et al]

correlation polytope: $P_{corr} = \mathrm{conv}\{aa^T | a \text{ in } \{0,1\}^n\}$

vertices:

constraints:



The Construction of [Fiorini et al]

correlation polytope: $P_{corr} = \text{conv}\{aa^T | a \text{ in } \{0,1\}^n\}$

vertices: a in $\{0,1\}^n$

constraints:

b in $\{0,1\}^n$

S

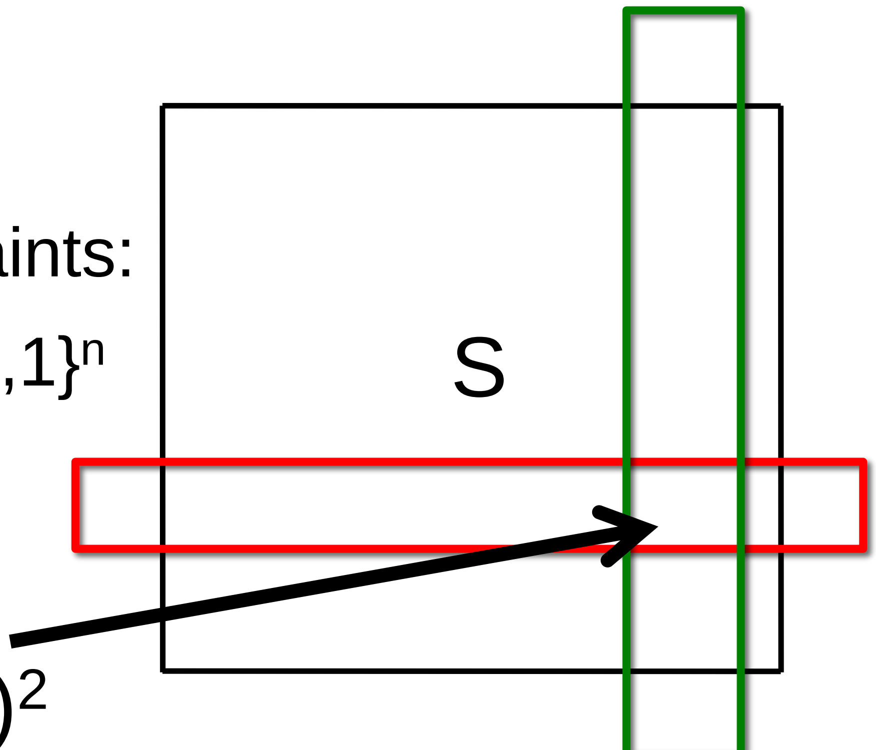
The Construction of [Fiorini et al]

correlation polytope: $P_{corr} = \text{conv}\{aa^T | a \text{ in } \{0,1\}^n\}$

vertices: a in $\{0,1\}^n$

constraints:

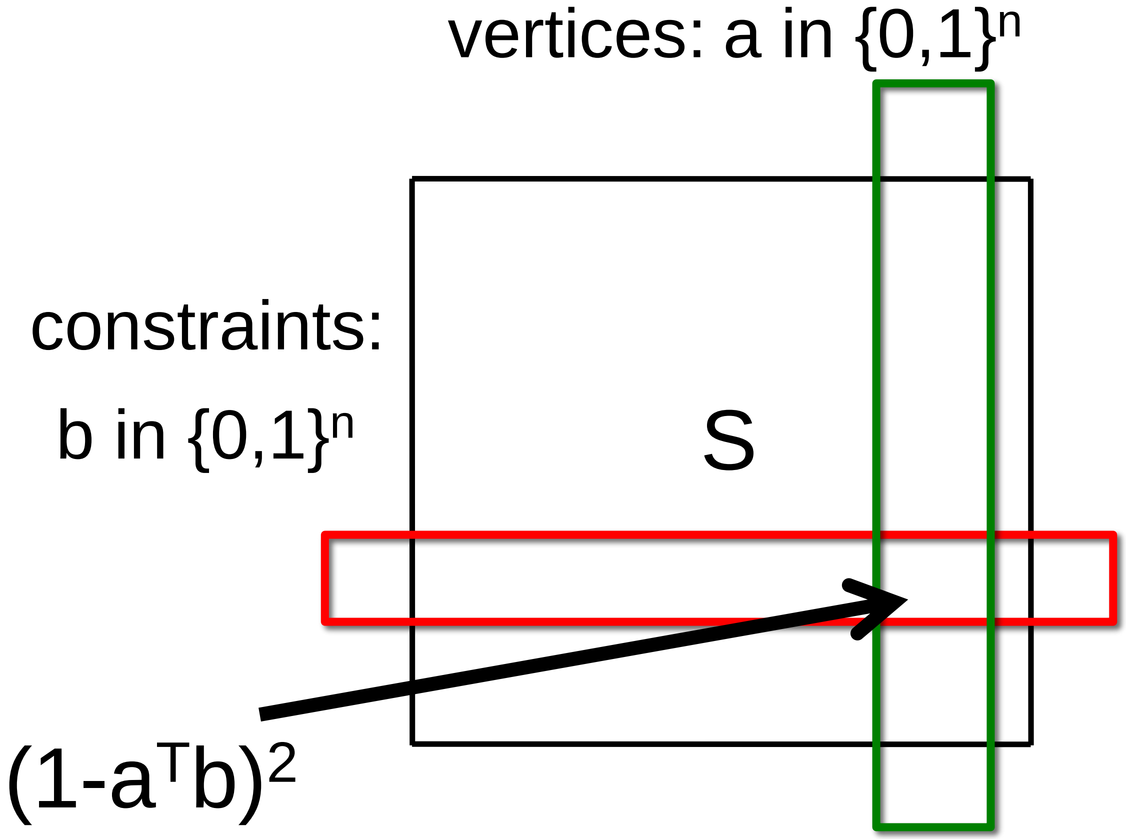
b in $\{0,1\}^n$



$(1-a^Tb)^2$

The Construction of [Fiorini et al]

correlation polytope: $P_{corr} = \text{conv}\{aa^T | a \text{ in } \{0,1\}^n\}$



UNIQUE DISJ.
Output 'YES' if a and b as sets are disjoint, and 'NO' if a and b have one index in common

The Reconstruction Principle

The Reconstruction Principle

Let $T = \{(a,b) \mid a^T b = 0\}$, $|T| = 3^n$

The Reconstruction Principle

Let $T = \{(a,b) \mid a^T b = 0\}$, $|T| = 3^n$

Recall: $S_{a,b}=(1-a^T b)^2$, so $S_{a,b}=1$ for all pairs in T

The Reconstruction Principle

Let $T = \{(a,b) \mid a^T b = 0\}$, $|T| = 3^n$

Recall: $S_{a,b}=(1-a^T b)^2$, so $S_{a,b}=1$ for all pairs in T

How does the sampling procedure **specialize** to this case? (Recall it generates (a,b) unif. from T)

The Reconstruction Principle

Let $T = \{(a,b) \mid a^T b = 0\}$, $|T| = 3^n$

Recall: $S_{a,b}=(1-a^T b)^2$, so $S_{a,b}=1$ for all pairs in T

How does the sampling procedure **specialize** to this case? (Recall it generates (a,b) unif. from T)

Sampling Procedure:

The Reconstruction Principle

Let $T = \{(a,b) \mid a^T b = 0\}$, $|T| = 3^n$

Recall: $S_{a,b}=(1-a^T b)^2$, so $S_{a,b}=1$ for all pairs in T

How does the sampling procedure **specialize** to this case? (Recall it generates (a,b) unif. from T)

Sampling Procedure:

- Let R_i be the sum of $M_i(a,b)$ over (a,b) in T and let R be the sum of R_i

The Reconstruction Principle

Let $T = \{(a,b) \mid a^T b = 0\}$, $|T| = 3^n$

Recall: $S_{a,b}=(1-a^T b)^2$, so $S_{a,b}=1$ for all pairs in T

How does the sampling procedure **specialize** to this case? (Recall it generates (a,b) unif. from T)

Sampling Procedure:

- Let R_i be the sum of $M_i(a,b)$ over (a,b) in T and let R be the sum $_{\text{of } R_i}$
- Choose i with probability R_i/R

The Reconstruction Principle

Let $T = \{(a,b) \mid a^T b = 0\}$, $|T| = 3^n$

Recall: $S_{a,b}=(1-a^T b)^2$, so $S_{a,b}=1$ for all pairs in T

How does the sampling procedure **specialize** to this case? (Recall it generates (a,b) unif. from T)

Sampling Procedure:

- Let R_i be the sum of $M_i(a,b)$ over (a,b) in T and let R be the sum $_{\text{of } R_i}$
- Choose i with probability R_i/R
- Choose (a,b) with probability $M_i(a,b)/R_i$

Entropy Accounting 101

Entropy Accounting 101

Sampling Procedure:

- Let R_i be the sum of $M_i(a,b)$ over (a,b) in T and let R be the sum of R_i
- Choose i with probability R_i/R
- Choose (a,b) with probability $M_i(a,b)/R_i$

Entropy Accounting 101

Sampling Procedure:

- Let R_i be the sum of $M_i(a,b)$ over (a,b) in T and let R be the sum of R_i
- Choose i with probability R_i/R
- Choose (a,b) with probability $M_i(a,b)/R_i$

Total Entropy:

$$n \log_2 3 \quad \leq$$

Entropy Accounting 101

Sampling Procedure:

- Let R_i be the sum of $M_i(a,b)$ over (a,b) in T and let R be the sum of R_i
- Choose i with probability R_i/R
- Choose (a,b) with probability $M_i(a,b)/R_i$

Total Entropy:

$$n \log_2 3 \leq \overbrace{\qquad\qquad}^{\text{choose } i} + \overbrace{\qquad\qquad\qquad\qquad}^{\text{choose } (a,b) \text{ conditioned on } i}$$

Entropy Accounting 101

Sampling Procedure:

- Let R_i be the sum of $M_i(a,b)$ over (a,b) in T and let R be the sum of R_i
- Choose i with probability R_i/R
- Choose (a,b) with probability $M_i(a,b)/R_i$

Total Entropy:

$$n \log_2 3 \leq \underbrace{\log_2 r}_{\text{choose } i} + \underbrace{\qquad\qquad}_{\text{choose } (a,b) \text{ conditioned on } i}$$

Entropy Accounting 101

Sampling Procedure:

- Let R_i be the sum of $M_i(a,b)$ over (a,b) in T and let R be the sum of R_i
- Choose i with probability R_i/R
- Choose (a,b) with probability $M_i(a,b)/R_i$

Total Entropy:

$$n \log_2 3 \leq \underbrace{\log_2 r}_{\text{choose } i} + \underbrace{(1-\delta) n \log_2 3}_{\text{choose } (a,b) \text{ conditioned on } i}$$

Entropy Accounting 101

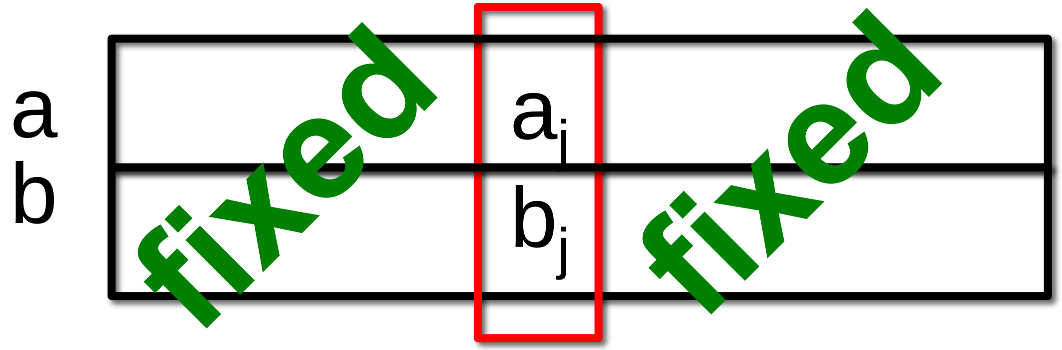
Sampling Procedure:

- Let R_i be the sum of $M_i(a,b)$ over (a,b) in T and let R be the sum of R_i
- Choose i with probability R_i/R
- Choose (a,b) with probability $M_i(a,b)/R_i$

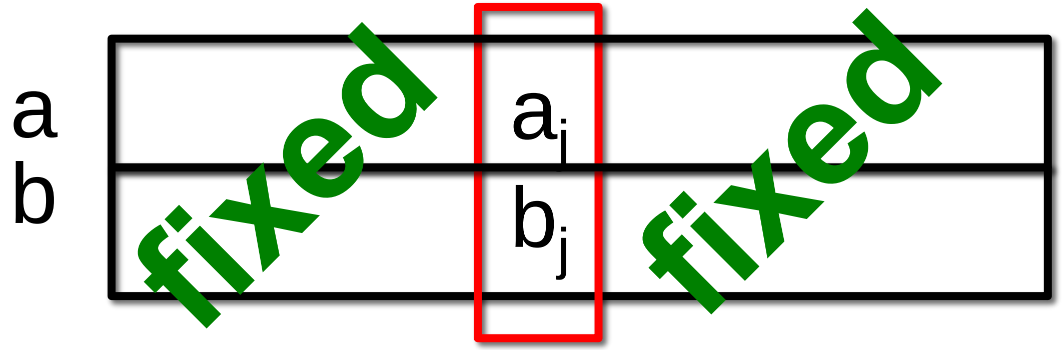
Total Entropy:

$$n \log_2 3 \leq \overbrace{\log_2 r}^{\text{choose } i} + \overbrace{(1-\delta) n \log_2 3}^{\text{choose } (a,b) \text{ conditioned on } i} \quad \textbf{(?)}$$

Suppose that a_{-j} and b_{-j} are **fixed**

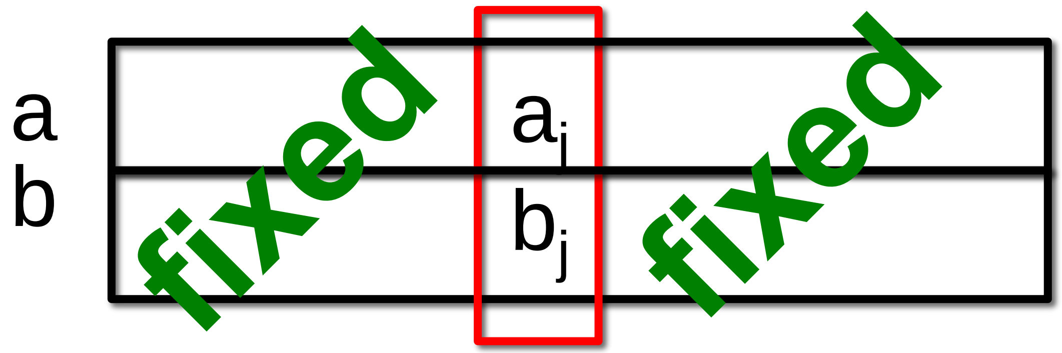


Suppose that a_{-j} and b_{-j} are **fixed**



M_i restricted to (a_{-j},b_{-j})

Suppose that a_{-j} and b_{-j} are **fixed**



M_i restricted to (a_{-j}, b_{-j})

	$(\ldots b_j=0\ldots)$	$(\ldots b_j=1\ldots)$
$(a_{1..j-1}, a_j=0, a_{j+1\ldots n})$	$M_i(a,b)$	$M_i(a,b)$
$(a_{1..j-1}, a_j=1, a_{j+1\ldots n})$	$M_i(a,b)$	$M_i(a,b)$

	$(\ldots b_j=0\ldots)$	$(\ldots b_j=1\ldots)$
$(a_{1..j-1},a_j=0,a_{j+1\ldots n})$	$M_i(a,b)$	$M_i(a,b)$
$(a_{1..j-1},a_j=1,a_{j+1\ldots n})$	$M_i(a,b)$	$M_i(a,b)$

If $a_j=1$, $b_j=1$ then $a^Tb = 1$, hence $M_i(a,b) = 0$

$$
\begin{array}{c|c|c|}
 & (\ldots b_j=0\ldots) & (\ldots b_j=1\ldots) \\
\hline
(a_{1..j-1},a_j=0,a_{j+1\ldots n}) & M_i(a,b) & M_i(a,b) \\
\hline
(a_{1..j-1},a_j=1,a_{j+1\ldots n}) & M_i(a,b) & M_i(a,b) \\
\hline
\end{array}
$$

If $a_j=1$, $b_j=1$ then $a^Tb = 1$, hence $M_i(a,b) = 0$

	$(\ldots b_j=0\ldots)$	$(\ldots b_j=1\ldots)$
$(a_{1..j-1},a_j=0,a_{j+1\ldots n})$	$M_i(a,b)$	$M_i(a,b)$
$(a_{1..j-1},a_j=1,a_{j+1\ldots n})$	$M_i(a,b)$	**zero**

If $a_j=1$, $b_j=1$ then $a^Tb = 1$, hence $M_i(a,b) = 0$

But $rank(M_i)=1$, hence there must be another zero in either the same row or column

	$(\ldots b_j=0\ldots)$	$(\ldots b_j=1\ldots)$
$(a_{1..j-1},a_j=0,a_{j+1\ldots n})$	$M_i(a,b)$	$M_i(a,b)$
$(a_{1..j-1},a_j=1,a_{j+1\ldots n})$	$M_i(a,b)$	**zero**

If $a_j=1$, $b_j=1$ then $a^Tb = 1$, hence $M_i(a,b) = 0$

But $\text{rank}(M_i)=1$, hence there must be another zero in either the same row or column

	$(\ldots b_j=0\ldots)$	$(\ldots b_j=1\ldots)$
$(a_{1..j-1},a_j=0,a_{j+1\ldots n})$	$M_i(a,b)$	$M_i(a,b)$
$(a_{1..j-1},a_j=1,a_{j+1\ldots n})$	**zero**	**zero**

If $a_j=1$, $b_j=1$ then $a^T b = 1$, hence $M_i(a,b) = 0$

But $\mathrm{rank}(M_i)=1$, hence there must be another zero in either the same row or column

$H(a_j,b_j \mid i, a_{-j}, b_{-j}) \leq 1 < \log_2 3$

	$(\ldots b_j=0\ldots)$	$(\ldots b_j=1\ldots)$
$(a_{1..j-1},a_j=0,a_{j+1\ldots n})$	$M_i(a,b)$	$M_i(a,b)$
$(a_{1..j-1},a_j=1,a_{j+1\ldots n})$	**zero**	**zero**

Entropy Accounting 101

Generate uniformly random (a,b) in T:

- Let R_i be the sum of $M_i(a,b)$ over (a,b) in T and let R be the sum of R_i
- Choose i with probability R_i/R
- Choose (a,b) with probability $M_i(a,b)/R_i$

Total Entropy:

$$n \log_2 3 \le \underbrace{\log_2 r}_{\text{choose } i} + \underbrace{\qquad\qquad\qquad}_{\text{choose } (a,b) \text{ conditioned on } i}$$

Entropy Accounting 101

Generate uniformly random (a,b) in T:

- Let R_i be the sum of $M_i(a,b)$ over (a,b) in T and let R be the sum of R_i
- Choose i with probability R_i/R
- Choose (a,b) with probability $M_i(a,b)/R_i$

Total Entropy:

$$n \log_2 3 \leq \underbrace{\log_2 r}_{\text{choose } i} + \underbrace{n}_{\text{choose } (a,b) \text{ conditioned on } i}$$

Outline

Part I: Tools for Extended Formulations

- Yannakakis's Factorization Theorem
- The Rectangle Bound
- A Sampling Argument

Part II: Applications

- Correlation Polytope
- Approximating the Correlation Polytope
- A Better Lower Bound for Disjointness

Outline

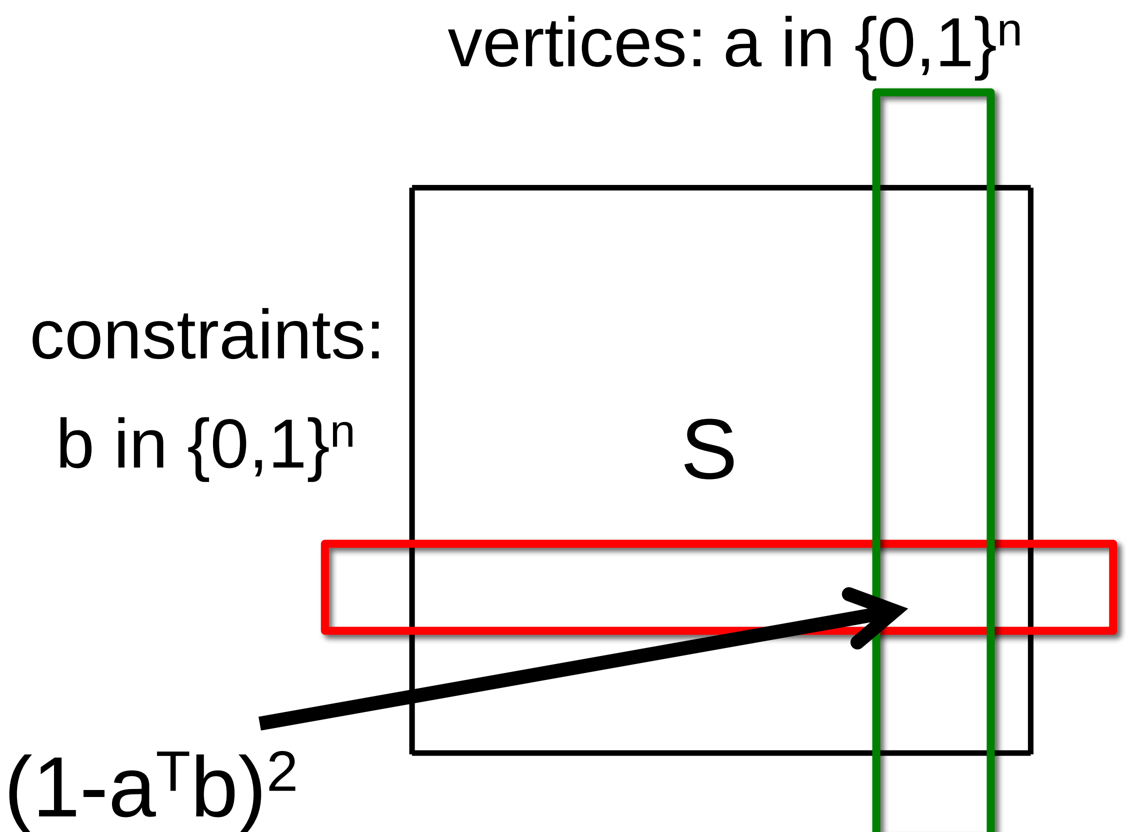
Part I: Tools for Extended Formulations

- Yannakakis's Factorization Theorem
- The Rectangle Bound
- A Sampling Argument

Part II: Applications

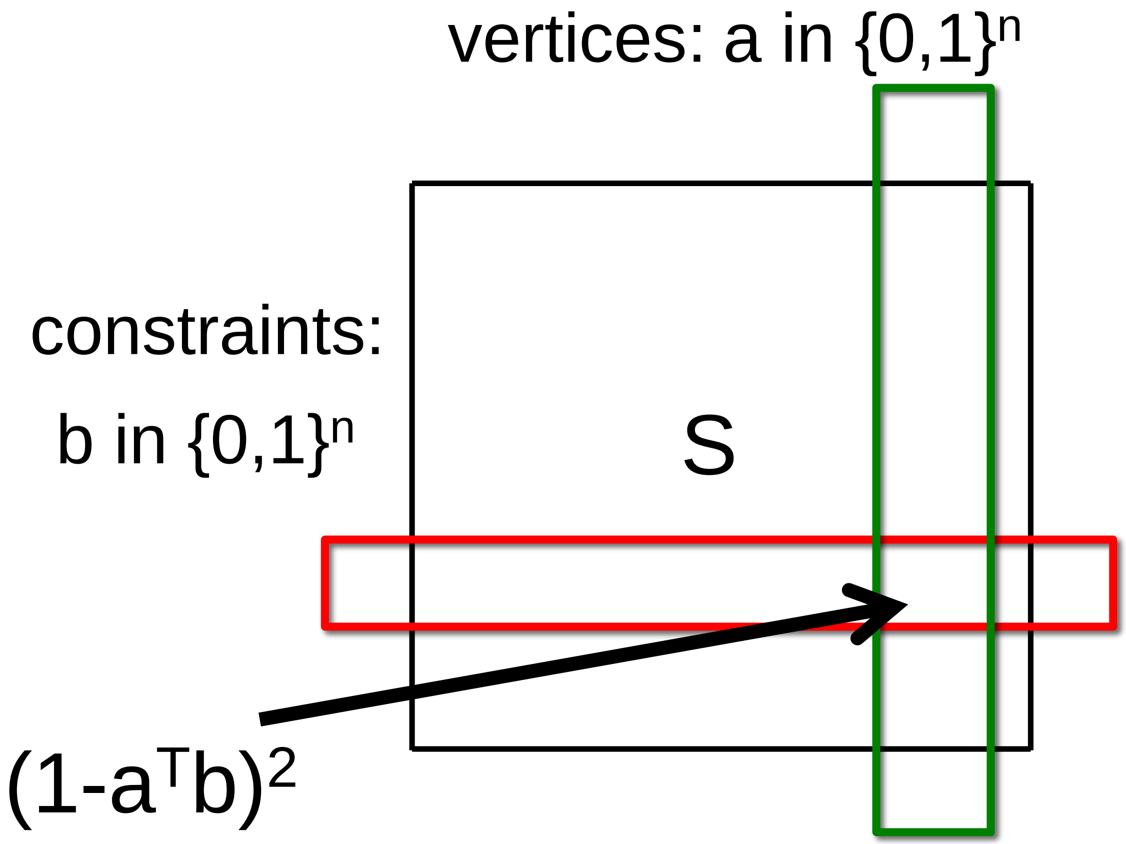
- Correlation Polytope
- **Approximating the Correlation Polytope**
- A Better Lower Bound for Disjointness

Approximate EFs [Braun et al]



Approximate EFs [Braun et al]

Is there a K (with small xc) s.t. $P_{corr} \subset K \subset (C+1)P_{corr}$?



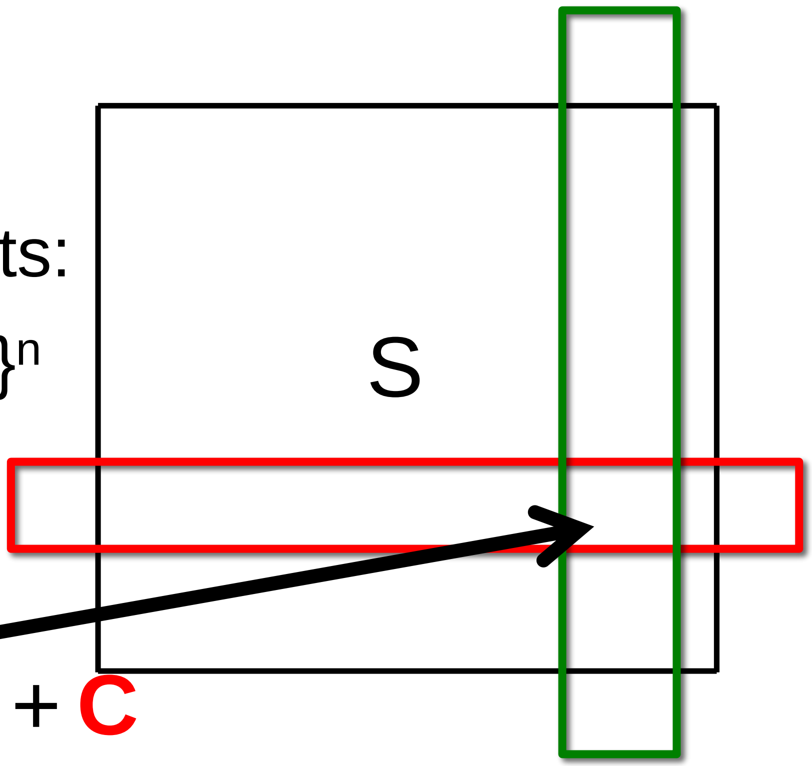
Approximate EFs [Braun et al]

Is there a K (with small xc) s.t. $P_{corr} \subset K \subset (C+1)P_{corr}$?

vertices: a in $\{0,1\}^n$

constraints:

b in $\{0,1\}^n$



$(1-a^Tb)^2$ + **C**

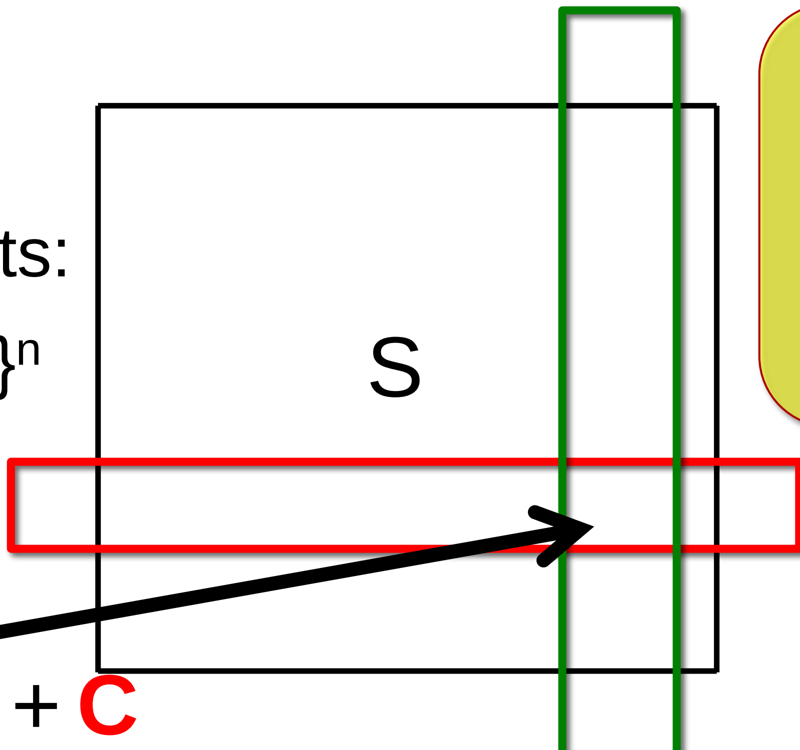
Approximate EFs [Braun et al]

Is there a K (with small xc) s.t. $P_{corr} \subset K \subset (C+1)P_{corr}$?

vertices: a in $\{0,1\}^n$

constraints:

b in $\{0,1\}^n$



$(1-a^Tb)^2$ + **C**

New Goal:
Output the answer to UDISJ with prob. at least ½ + 1/2(C+1)

Is the correlation polytope hard to approximate for large values of C?

Analogy: Is UDISJ hard to compute with prob. $\frac{1}{2}+1/2(C+1)$ for large values of C?

Is the correlation polytope hard to approximate for large values of C?

Analogy: Is UDISJ hard to compute with prob. $\frac{1}{2}+1/2(C+1)$ for large values of C?

There is a natural barrier at $C = \sqrt{n}$ for proving l.b.s:

Is the correlation polytope hard to approximate for large values of C?

Analogy: Is UDISJ hard to compute with prob. ½+1/2(C+1) for large values of C?

There is a natural barrier at $C = \sqrt{n}$ for proving l.b.s:

Claim: If UDISJ can be computed with prob. ½+1/2(C+1) using $o(n/C^2)$ bits, then UDISJ can be computed with prob. ¾ using $o(n)$ bits

Is the correlation polytope hard to approximate for large values of C?

Analogy: Is UDISJ hard to compute with prob. ½+1/2(C+1) for large values of C?

There is a natural barrier at $C = \sqrt{n}$ for proving l.b.s:

Claim: If UDISJ can be computed with prob. ½+1/2(C+1) using $o(n/C^2)$ bits, then UDISJ can be computed with prob. ¾ using o(n) bits

Proof: Run the protocol $O(C^2)$ times and take the majority vote

Information Complexity

Information Complexity

In fact, a more technical barrier is:

Information Complexity

In fact, a more technical barrier is:

[Bar-Yossef et al ’04]:

Bits exchanged $\geq$ information revealed

Information Complexity

In fact, a more technical barrier is:

[Bar-Yossef et al ’04]:

Bits exchanged ≥ information revealed

≥ n x information revealed
for a one-bit problem

Information Complexity

In fact, a more technical barrier is:

[Bar-Yossef et al ’04]:

Bits exchanged ≥ information revealed

Direct Sum Theorem

≥ n x information revealed
for a one-bit problem

Information Complexity

In fact, a more technical barrier is:

[Bar-Yossef et al ’04]:

Bits exchanged ≥ information revealed

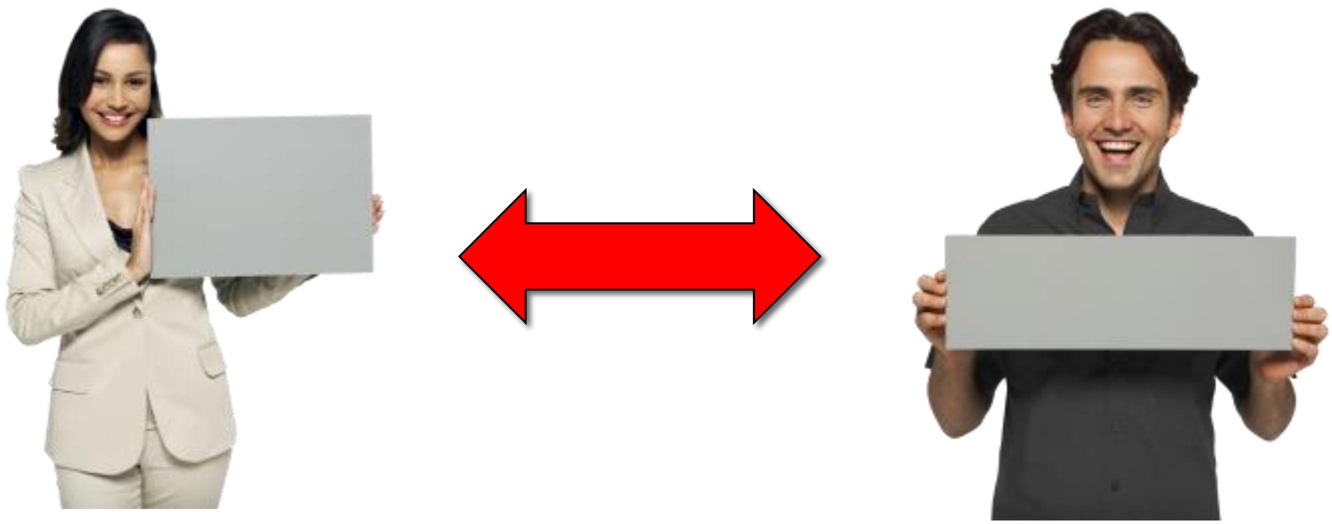
Direct Sum Theorem

≥ n x information revealed
for a one-bit problem

Problem: AND has a protocol with advantage $1/C$ that reveals only $1/C^2$ bits…

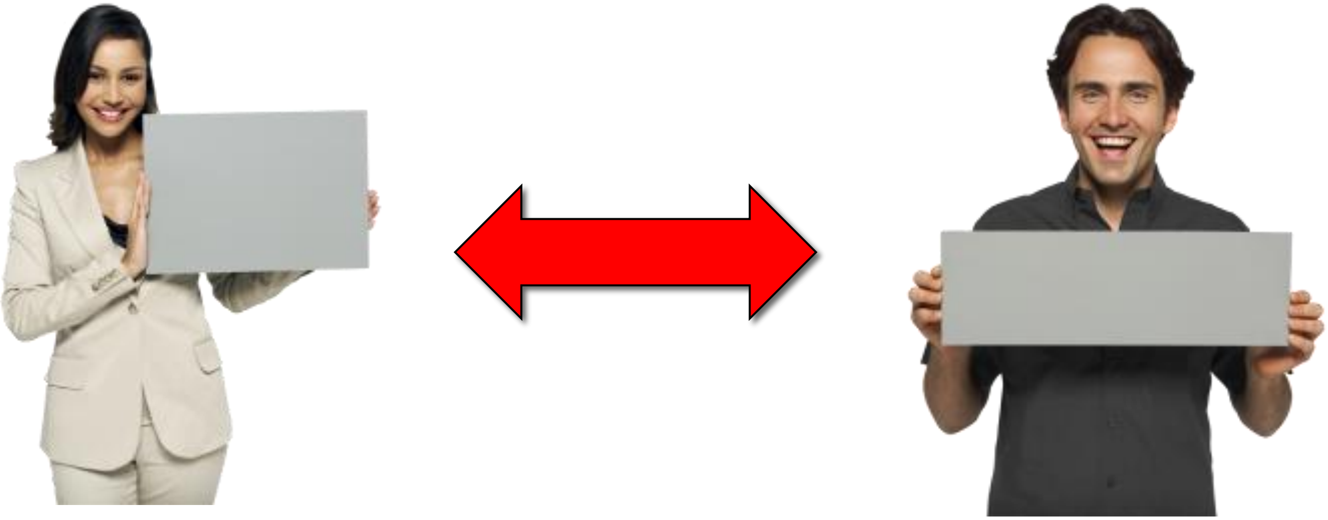
Problem: AND has a protocol with advantage $1/C$ that reveals only $1/C^2$ bits…

Problem: AND has a protocol with advantage 1/C that reveals only $1/C^2$ bits…



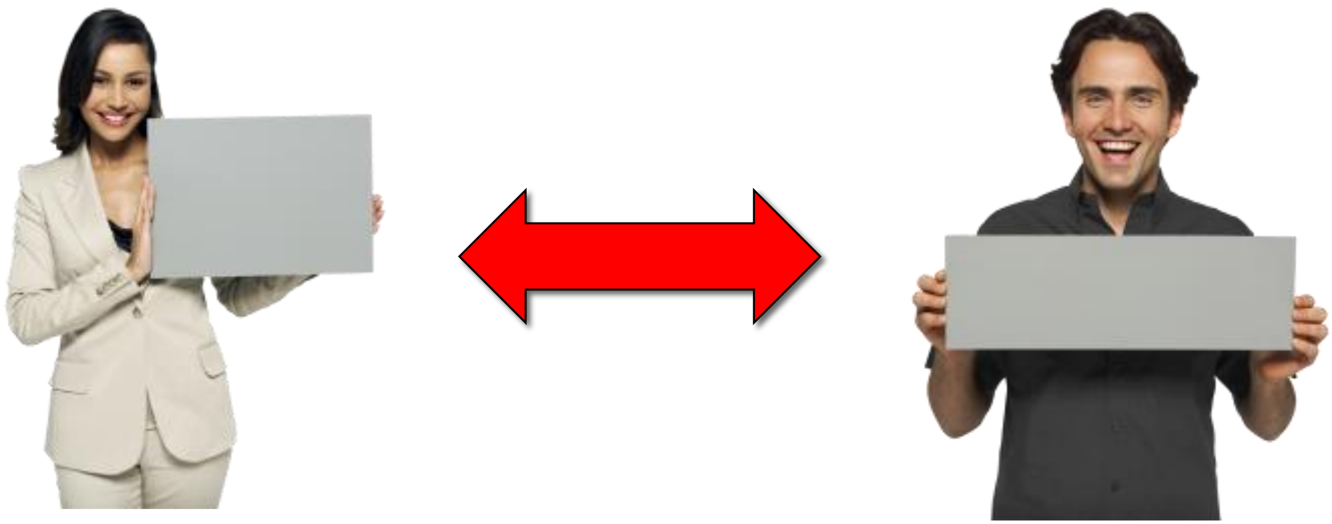
Send her bit with prob.
½ + 1/C, else send
the complement

Problem: AND has a protocol with advantage 1/C that reveals only $1/C^2$ bits…



Send her bit with prob. ½ + 1/C, else send the complement

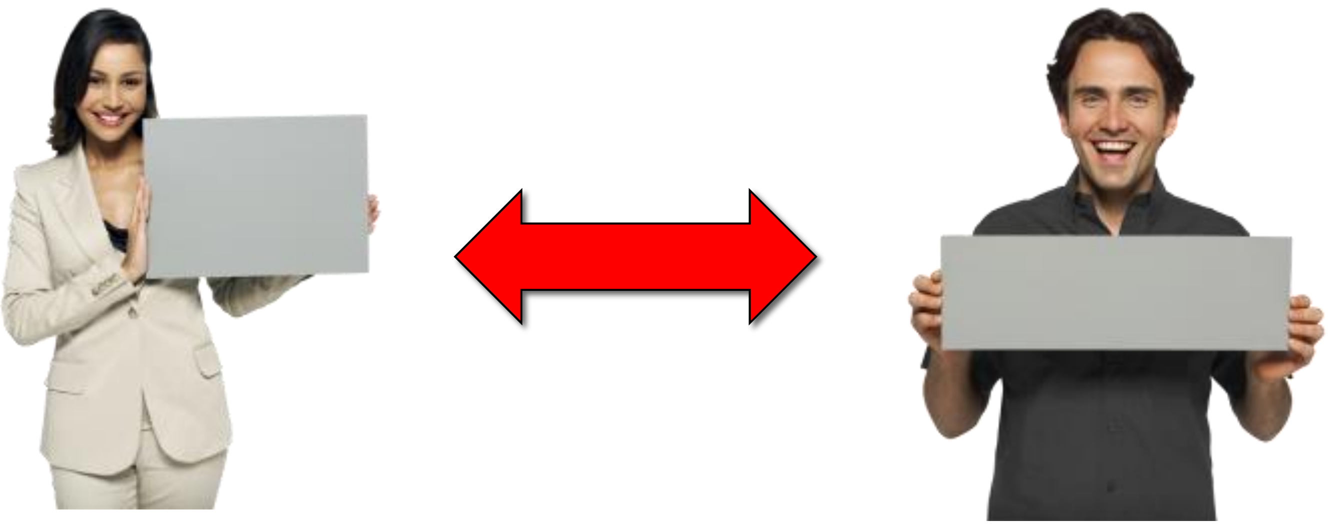
Send his bit with prob. ½ + 1/C, else send the complement



Send her bit with prob. ½ + 1/C, else send the complement

Send his bit with prob. ½ + 1/C, else send the complement

Is there a **stricter** one bit problem that we could reduce to instead?



Send her bit with prob. ½ + 1/C, else send the complement

Send his bit with prob. ½ + 1/C, else send the complement

Outline

Part I: Tools for Extended Formulations

- Yannakakis's Factorization Theorem
- The Rectangle Bound
- A Sampling Argument

Part II: Applications

- Correlation Polytope
- Approximating the Correlation Polytope
- A Better Lower Bound for Disjointness

Outline

Part I: Tools for Extended Formulations

- Yannakakis's Factorization Theorem
- The Rectangle Bound
- A Sampling Argument

Part II: Applications

- Correlation Polytope
- Approximating the Correlation Polytope
- **A Better Lower Bound for Disjointness**

Definition: Output matrix is the prob. of outputting "one" for each pair of inputs

Definition: Output matrix is the prob. of outputting "one" for each pair of inputs

For the previous protocol:

	B = 0	B = 1
A = 0	$\frac{1}{2} + 5/C$	$\frac{1}{2} + 1/C$
A = 1	$\frac{1}{2} + 1/C$	$\frac{1}{2} - 3/C$

The constraint that a protocol achieves advantage at least 1/C is a set of linear constraints on this matrix

For the previous protocol:

	B = 0	B = 1
A = 0	½ + 5/C	½ + 1/C
A = 1	½ + 1/C	½ - 3/C

The constraint that a protocol achieves advantage at least 1/C is a set of linear constraints on this matrix

For the previous protocol:

	B = 0	B = 1
A = 0	½ + 5/C	½ + 1/C
A = 1	½ + 1/C	½ - 3/C

(Using Hellinger): bits revealed $\geq$ (max– min)^2 = $\Omega(1/C^2)$

The constraint that a protocol achieves advantage at least 1/C is a set of linear constraints on this matrix

For the previous protocol:

	B = 0	B = 1
A = 0	½ + 5/C	½ + 1/C
A = 1	½ + 1/C	½ - 3/C

(Using Hellinger): bits revealed ≥ $(\max - \min)^2 = \Omega(1/C^2)$

(New): bits revealed ≥ |diagonal – anti-diagonal| = 0

For the previous protocol:

	B = 0	B = 1
A = 0	½ + 5/C	½ + 1/C
A = 1	½ + 1/C	½ - 3/C

What if we also require the output distribution to be the same for inputs {0,0}, {0,1}, {1,0}?

For the previous protocol:

	B = 0	B = 1
A = 0	$\frac{1}{2} + 5/C$	$\frac{1}{2} + 1/C$
A = 1	$\frac{1}{2} + 1/C$	$\frac{1}{2} - 3/C$

What if we also require the output distribution to be the same for inputs {0,0}, {0,1}, {1,0}?

For the ~~previous~~ **new** protocol:

	B = 0	B = 1
A = 0	½ + 1/C	½ + 1/C
A = 1	½ + 1/C	½ - 3/C

What if we also require the output distribution to be the same for inputs {0,0}, {0,1}, {1,0}?

For the ~~previous~~ **new** protocol:

	B = 0	B = 1
A = 0	½ + 1/C	½ + 1/C
A = 1	½ + 1/C	½ - 3/C

(Using Hellinger): bits revealed ≥ $(\max - \min)^2 = \Omega(1/C^2)$

What if we also require the output distribution to be the same for inputs {0,0}, {0,1}, {1,0}?

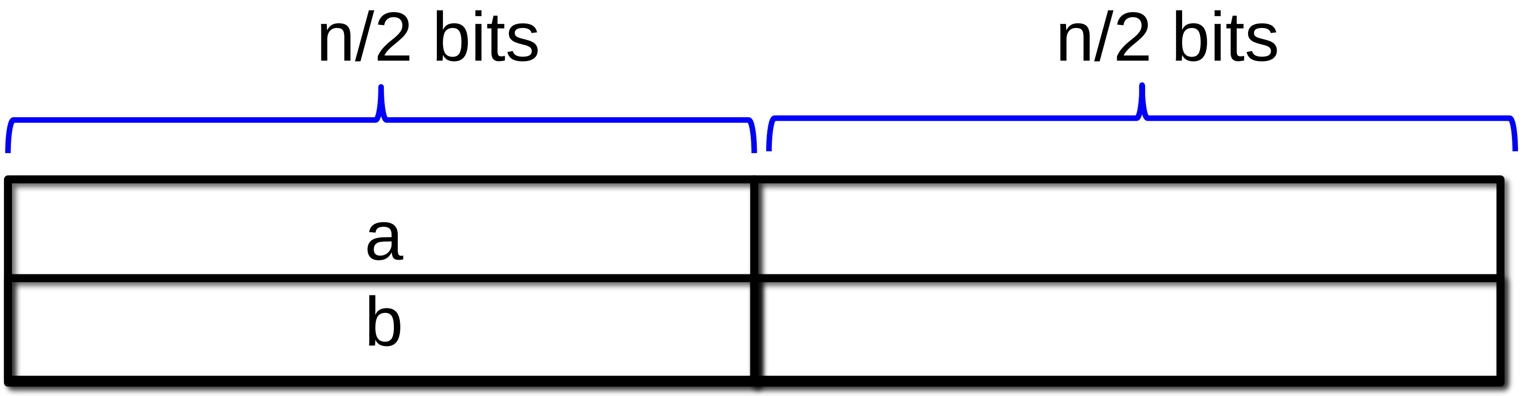
For the ~~previous~~ **new** protocol:

	B = 0	B = 1
A = 0	½ + 1/C	½ + 1/C
A = 1	½ + 1/C	½ - 3/C

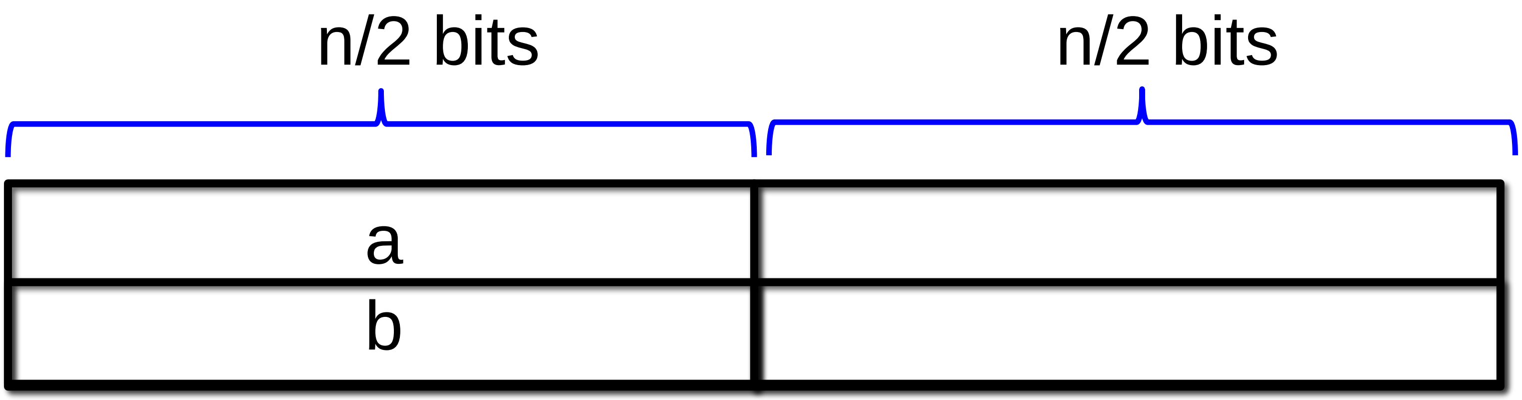
(Using Hellinger): bits revealed ≥ $(\max - \min)^2 = \Omega(1/C^2)$

(New): bits revealed ≥ |diagonal – anti-diagonal| = $\Omega(1/C)$

How can we reduce to such a one-bit problem?

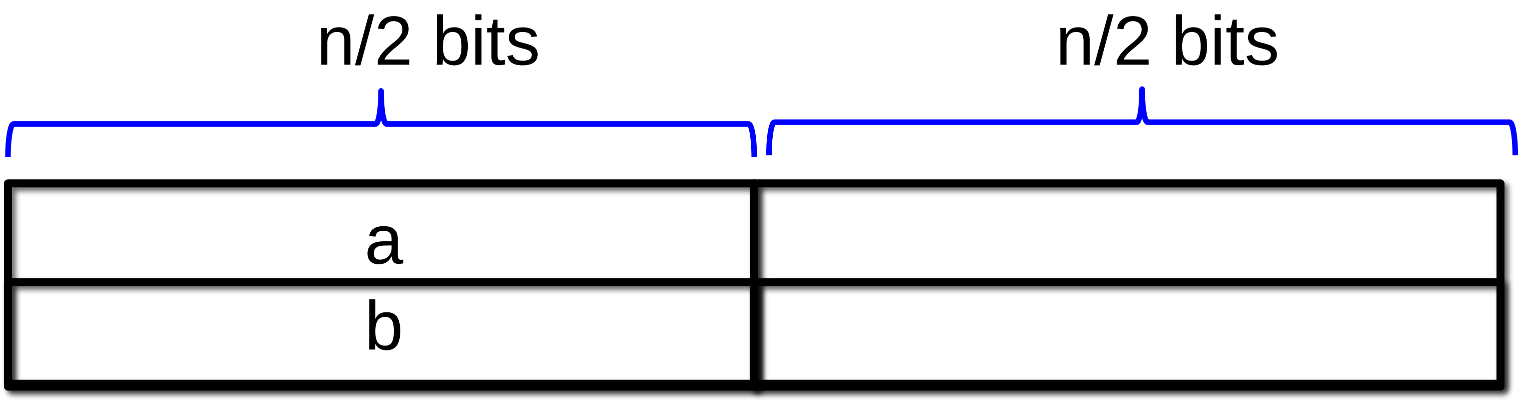


How can we reduce to such a one-bit problem?



Symmetrized Protocol T’:

How can we reduce to such a one-bit problem?



Symmetrized Protocol T':

- Generate a random partition (x, y, z) of n/2 and fill in x (0,0), y (0,1) and z (1,0) pairs

How can we reduce to such a one-bit problem?

n/2 bits	n/2 bits
a	0 … 0 … 1 …
b	0 … 1 … 0 …

Symmetrized Protocol T’:

- Generate a random partition (x, y, z) of n/2 and fill in x (0,0), y (0,1) and z (1,0) pairs

How can we reduce to such a one-bit problem?

n/2 bits	n/2 bits
a	0 … 0 … 1 …
b	0 … 1 … 0 …

Symmetrized Protocol T':

- Generate a random partition (x, y, z) of n/2 and fill in x (0,0), y (0,1) and z (1,0) pairs
- Permute the n bits uniformly at random

How can we reduce to such a one-bit problem?

n/2 bits	n/2 bits					
a	0	…	0	…	1	…
b	0	…	1	…	0	…

Symmetrized Protocol T':

- Generate a random partition (x, y, z) of n/2 and fill in x (0,0), y (0,1) and z (1,0) pairs
- Permute the n bits uniformly at random
- Run T on the n bit string, return the output

Symmetrized Protocol T':

- Generate a random partition (x, y, z) of n/2 and fill in x (0,0), y (0,1) and z (1,0) pairs
- Permute the n bits uniformly at random
- Run T on the n bit string, return the output

Symmetrized Protocol T':

- Generate a random partition (x, y, z) of n/2 and fill in x (0,0), y (0,1) and z (1,0) pairs
- Permute the n bits uniformly at random
- Run T on the n bit string, return the output

Claim: The protocol T' has bias 1/C for DISJ.

Symmetrized Protocol T':

- Generate a random partition (x, y, z) of n/2 and fill in x (0,0), y (0,1) and z (1,0) pairs
- Permute the n bits uniformly at random
- Run T on the n bit string, return the output

Claim: The protocol T' has bias 1/C for DISJ.

Claim: Yet flipping a pair (a_i, b_i) between (0,0), (0,1) or (1,0) results in two distributions p,q (on inputs to T) with $|p-q|_1 \leq 1/n$

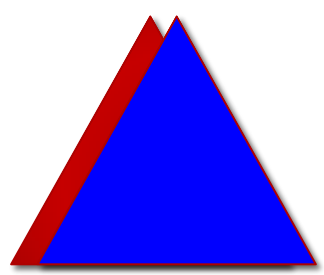
Symmetrized Protocol T':

- Generate a random partition (x, y, z) of n/2 and fill in x (0,0), y (0,1) and z (1,0) pairs
- Permute the n bits uniformly at random
- Run T on the n bit string, return the output

Claim: The protocol T' has bias 1/C for DISJ.

Claim: Yet flipping a pair (a_i, b_i) between (0,0), (0,1) or (1,0) results in two distributions p,q (on inputs to T) with $|p-q|_1 \leq 1/n$

Proof:



Theorem: any protocol for UDISJ with adv. $1/C$ must reveal $\Omega(n/C)$ bits (because it can be symmetrized)

Theorem: any protocol for UDISJ with adv. 1/C must reveal $\Omega(n/C)$ bits (because it can be symmetrized)

Similarly sampling a pair (a_j, b_j) needs entropy $\log_2 3 - \delta$, where δ is the **difference of diagonals**

Theorem: any protocol for UDISJ with adv. 1/C must reveal $\Omega(n/C)$ bits (because it can be symmetrized)

Similarly sampling a pair (a_j, b_j) needs entropy $\log_2 3 - \delta$, where δ is the **difference of diagonals**

Theorem: For any K with $P_{corr} \subset K \subset (C+1)P_{corr}$, the extension complexity of K is at least $\exp(\Omega(n/C))$

Theorem: any protocol for UDISJ with adv. 1/C must reveal $\Omega(n/C)$ bits (because it can be symmetrized)

Similarly sampling a pair (a_j, b_j) needs entropy $\log_2 3 - \delta$, where δ is the **difference of diagonals**

Theorem: For any K with $P_{corr} \subset K \subset (C+1)P_{corr}$, the extension complexity of K is at least $\exp(\Omega(n/C))$

Can our framework be used to prove further lower bounds for extension complexity?

Theorem: any protocol for UDISJ with adv. $1/C$ must reveal $\Omega(n/C)$ bits (because it can be symmetrized)

Similarly sampling a pair (a_j, b_j) needs entropy $\log_2 3 - \delta$, where δ is the **difference of diagonals**

Theorem: For any K with $P_{corr} \subset K \subset (C+1)P_{corr}$, the extension complexity of K is at least $\exp(\Omega(n/C))$

Can our framework be used to prove further lower bounds for extension complexity?

For average case instances?

Theorem: any protocol for UDISJ with adv. 1/C must reveal $\Omega(n/C)$ bits (because it can be symmetrized)

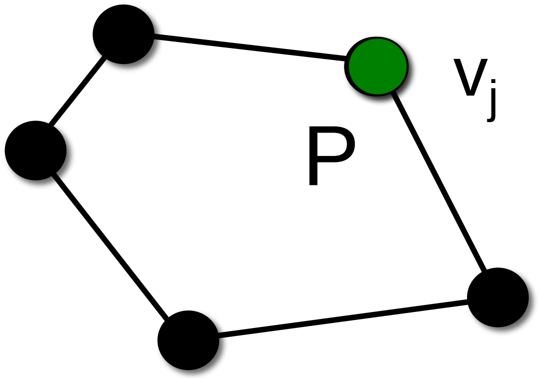
Similarly sampling a pair (a_j, b_j) needs entropy $\log_2 3 - \delta$, where δ is the **difference of diagonals**

Theorem: For any K with $P_{corr} \subset K \subset (C+1)P_{corr}$, the extension complexity of K is at least $\exp(\Omega(n/C))$

Can our framework be used to prove further lower bounds for extension complexity?

For average case instances? For SDPs?

Thanks!



Any Questions?