An introduction to Extended Formulations - capturing the expressive power of LPs and SDPs

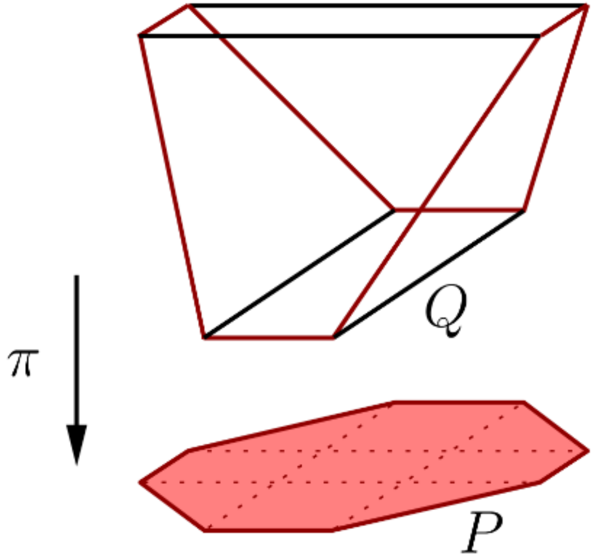
Sebastian Pokutta
Georgia Institute of Technology
School of Industrial and Systems Engineering (ISyE)
Algorithms and Randomness Center (ARC)

Optimization Without Borders
Dedicated to Yuri Nesterov's 60th Birthday
Les Houches, 2/2016

Georgia Tech

Introduction and Motivation

Projections can drastically increase the complexity of a description.



Can we invert the process to find small formulations for complicated problems?

Introduction and Motivation

This phenomenon is well known (but not understood):

1. Quantifier elimination
2. The quest of P vs. NP is exactly of this type: $\boldsymbol{x \in L \Leftrightarrow \exists y: f(x, y) = 1,}$
3. Fourier-Motzkin elimination leads to exponential blow-up

Extended formulations = quantifier elimination backwards

Natural questions.

1. Maybe *any* 0/1 polytope has poly-size extended formulations?
2. ∃ problems in P, however do not admit small formulations?
3. ∃ problems that admit good approximations via small SDPs but not LPs?

Georgia Tech

Introduction and Motivation

How it all started…

In 86/87, Swart claimed he could prove $\boldsymbol{P = NP}$

How?

By giving a (purported) **poly-size linear program for the TSP problem**.

Theorem. [Yannakakis 88/91] Every symmetric LP for the TSP has size $2^{\Omega(n)}$.

Swart's LP was symmetric and of size $poly(n)$ => it was wrong.

Introduction and Motivation

However, that was not the end but the beginning.

1. [Kaibel, Pashkovich, Theis 10] symmetry can make a huge difference.
2. [Yannakakis 11] (20 years after his initial proof):

> I believe in fact that it should be possible to prove that there is no polynomial-size formulation for the TSP polytope or any other NP-hard problem, although of course showing this remains a challenging task.

3. Huge interest in finally ruling out all LPs of polynomial size for TSP.
4. [Fiorini, Massar, P., Tiwary, de Wolf 12] The TSP polytope has no small LPs.

The notion of LP-complexity (#inequalities) is **independent** of P vs. NP.
=> very strong indications for P vs. NP

Georgia Tech

Problems and LPs

Disclaimer. Similar for SDPs, however for simplicity confine to LPs.

Approximation Problems

An **approximation problem P** (max or min problem):

S: set of feasible solutions

F: set of considered objective functions (for simplicity: nonnegative)

κ: completeness guarantee, $\kappa(f) \in \mathbb{R}$ for each $f \in F$

τ: soundness guarantee, $\tau(f) \in \mathbb{R}$ for each $f \in F$

Goal. Whenever $f \in F$ with $\max_{s \in S} f(s) \leq \tau(f)$

Find: approximate solution with val $\leq \kappa(f)$ (max problem)

Example (exact min Vertex Cover): Given a graph G

S: all vertex covers of graph G (i.e., subsets of nodes covering all edges)

F: all nonnegative weight vectors on vertices

κ, τ: define $\kappa(f) = \tau(f) \coloneqq \min_{s \in S} f(s)$

Georgia Tech

LPs capturing Approximation Problems

Model of [Chan, Lee, Raghavendra, Steurer 13] and [Braun, P., Zink 14]

An **LP formulation of an approximation problem $P = (S, F, \kappa, \tau)$** is an LP $Ax \leq b$ with $x \in \mathbb{R}^d$ and *realizations*, where $F_\tau = \{f \in F \mid \max f(s) \leq \tau(f)\}$

a) *Feasible solutions:* for every $s \in S$ we have $x^s \in \mathbb{R}^d$ with

$Ax^s \leq b$ for all $s \in S$, (relaxation $conv(x^s \mid s \in S)$)

b) *Objective functions:* for every $f \in F_\tau$ we have an affine $w^f : \mathbb{R}^d \to \mathbb{R}$ with

$w^f(x^s) = f(s)$ for all $s \in S$, (linearization that is exact on S)

c) *Achieving* (κ, τ)-*approximation:* for every $f \in F_\tau$

$$\hat{f} = \max\{w^f(x) \mid Ax \leq b\} \leq \kappa(f)$$

Georgia Tech

Formulation Complexity

Approximation Problem $(\boldsymbol{S}, \boldsymbol{F}, \boldsymbol{\kappa}, \boldsymbol{\tau})$ → Slack matrix of problem $M_\tau\,(f, s) = \kappa(f) - f(s)$ → LP factorization $M_\tau = T \cdot U + \mu \cdot \mathbf{1}$ (restr. NMF)

Factorization theorem. Let $P = (S, F, \kappa, \tau)$ be a problem and M slack matrix of P

$$fc(P) = rank_{LP}(M_\tau)$$

Formulation complexity.

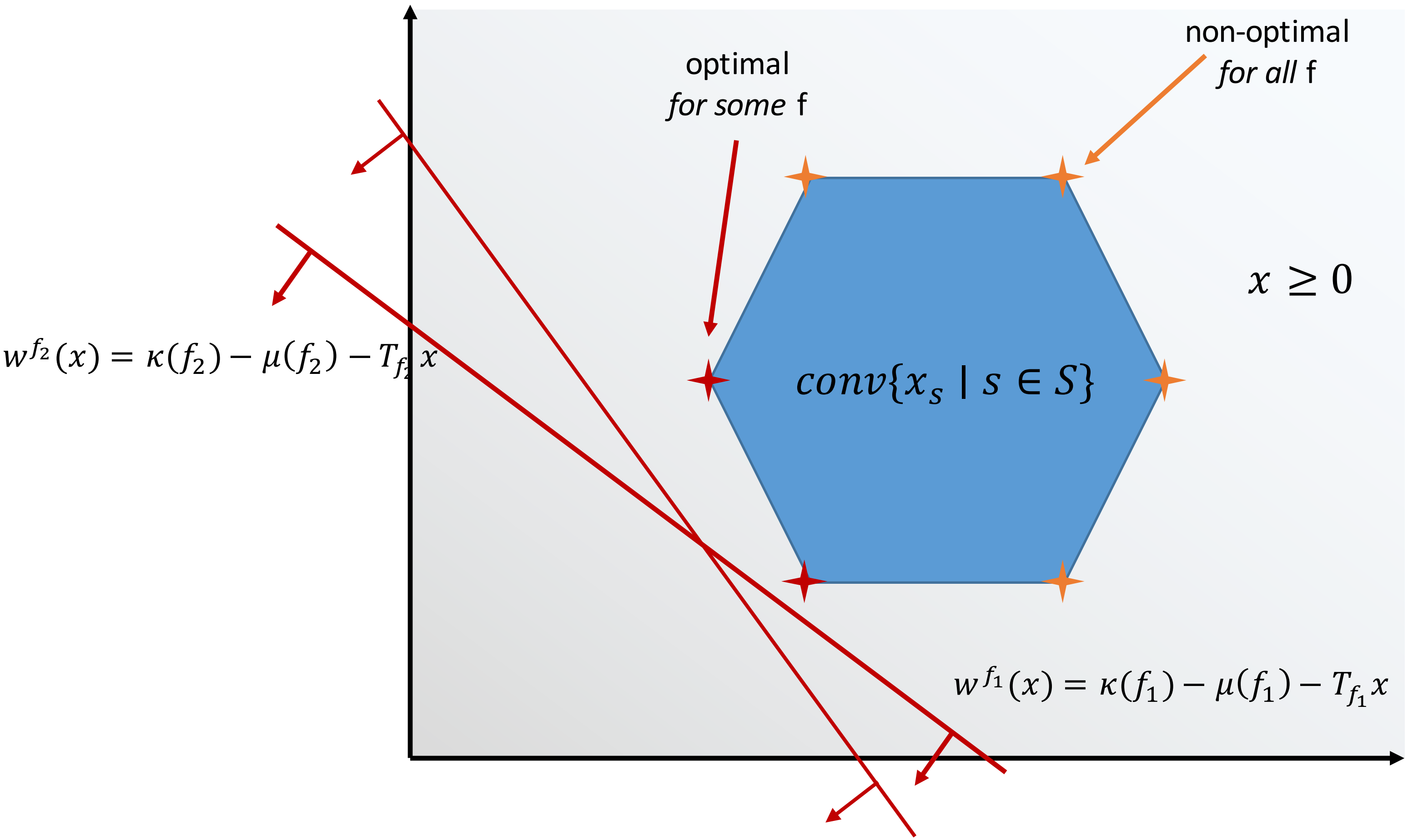
- Independent of P vs. NP
- Independent of a specific polyhedral representation
- Do not lift given representation but *construct* the optimal LP from factorization
- In fact: LP is trivial. Construct optimal encoding from factorization
- Restricted notion of nonnegative matrix factorization to support approximations

Optimal LP. $x \geq 0$ with encodings

feasible solutions: $x^s := U_s$ **objective functions:** $w^f(x) := \kappa(f) - \mu(f) - T_f \cdot x$

Georgia Tech

Optimal LPs

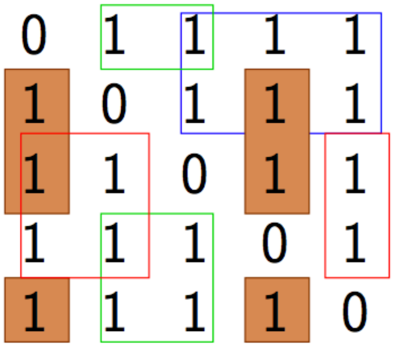


optimal
for some f
non-optimal
for all f
$x \geq 0$
$conv\{x_s \mid s \in S\}$
$w^{f_2}(x) = \kappa(f_2) - \mu(f_2) - T_{f_2} x$
$w^{f_1}(x) = \kappa(f_1) - \mu(f_1) - T_{f_1} x$

Lower bounding techniques

A simple lower bounding technique

The rectangle covering bound.



Consider hypothetical

$S = TU$ (nonnegative rank-r factorization)

$= \sum_{k=1,\ldots,r} T^k U_k$ (sum of r nonneg. rank-1 matrices)

Take support

$$\mathrm{supp}(S) = \bigcup_{k=1,\ldots,r} \mathrm{supp}\left(T^k U_k\right)$$

$$= \bigcup_{k=1,\ldots,r} \mathrm{supp}\left(T^k\right) \times \mathrm{supp}(U_k)$$ (union of r rectangles)

Rectangle covering number. $\mathrm{rc}(M) = \min\{r \mid \exists\ r\text{-size cover of } \mathrm{supp}(M)\}$

$$\Rightarrow \mathrm{rk}_+(M) \geq \mathrm{rc}(M)$$

The correlation polytope – an example

[Razborov 92] established in the context of nondeterministic communication:

Theorem. $rc(\mathrm{UDISJ_n}) = 2^{\varepsilon n}$

This implies [Fiorini, Massar, P., Tiwary, de Wolf 12]

$$2^{\varepsilon n} = rc(\mathrm{UDISJ_n}) \leq rk_+(\mathrm{UDISJ_n}) \leq rk_+(M_n) \leq \mathrm{fc}(\mathrm{COR}(n))$$

Hardness of approximation via

1. Communication complexity [Braun, Fiorini, P., Steurer 12] and
2. Information Theory [Braverman, Moitra 13] and [Braun, P. 13].

Theorem. Any LP approximating $\mathrm{COR}(n)$ within a factor $n^{1-\varepsilon}$ is of size $2^{\Theta(1)\varepsilon n}$.

Georgia Tech

The matching problem – a much more complicated case

Via a generalization of Razborov's technique:

Theorem. [Rothvoss 14] Any LP formulation of the matching polytope is of exponential size.

This is very special and important:

1. Matching can be solved in polynomial time
2. Yet any LP capturing it is of exponential size

=> Separates the power of P from polynomial size LPs

With information theory: ruling out the existence of FPTAS-type LP formulations

Theorem. [Braun, P. 14] For some $\varepsilon > 0$ any LP approximating the Matching Polytope within a factor $1 + \frac{\varepsilon}{n}$ is of exponential size.

Georgia Tech

Recent results for SDP extended formulations

Why are SDP EFs so much harder to understand?

… because they are so much stronger:

1. [Braun, Fiorini, P., Steurer 12]: ∃ (bounded) spectrahedron
   1. in dimension n^2, i.e., small SDP-EF
   2. any LP that approximates it within a factor of $n^{1-\varepsilon}$ is of size $2^{\Omega(n)}$
2. [Chain, Lee, Raghavendra, Steurer 13]: Separation via MaxCut
   1. The Goemans-Williams SDP gives an approximation of 0.87
   2. No polynomial-size LP can do better than 0.5
3. [Yannakakis 91]: Stable set polytope over perfect graphs
   1. Basic SDP gives perfect EF of the problem
   2. No polynomial size LP is known; best known $n^{O(\log n)}$

Georgia Tech

Known SDP EF lower bounds

Recent SDP EF lower bounds:

1. [Brïet, Dadush, P. 13]: Via counting argument
   1. There exist 0/1 polytopes that do not admit poly-size SDP EFs
   2. However, argument is only existential in nature

2. [Lee, Raghavendra, Steurer 14]: Bounds via (quantum) learning
   1. Reuse Lasserre gap instances and lower bounds
   2. Show that a hypothetical small SDP EF can be used to learn a good small Lasserre based SDP EF -> contradiction

3. [Braun, Brown-Cohen, Huq, P., Raghavendra, Roy, Weitz, Zink 15]: Y. for SDPs
   1. Matching has no small symmetric SDPs
   2. Among all symmetric SDPs of size $O(n^k)$ for TSP k-level Lasserre are best

4. No other direct / explicit lower bounding techniques are known

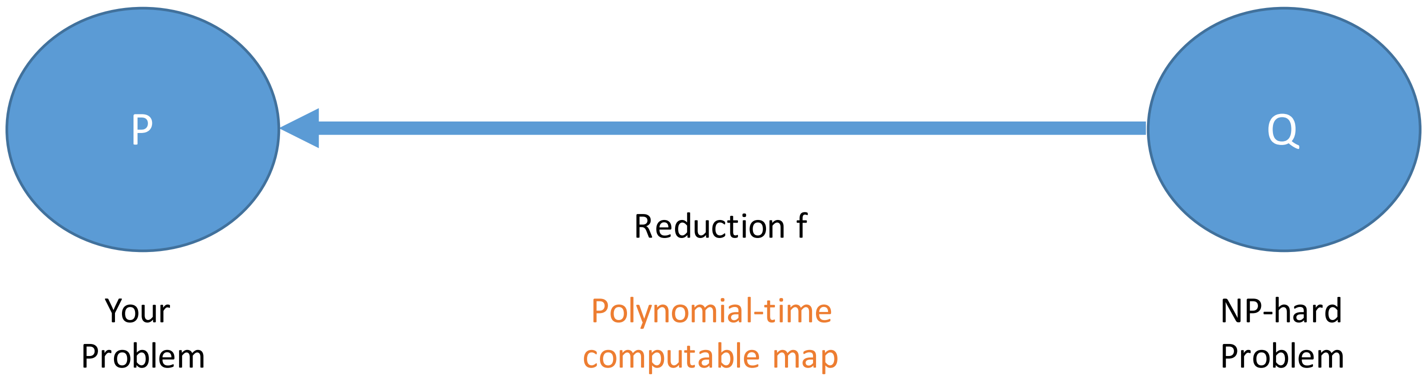
Georgia Tech

Reductions for
for extended formulations

(reuse what you know)

Reductions between Problems

In complexity theory. How to show that a problem P is NP-hard (to approximate)?

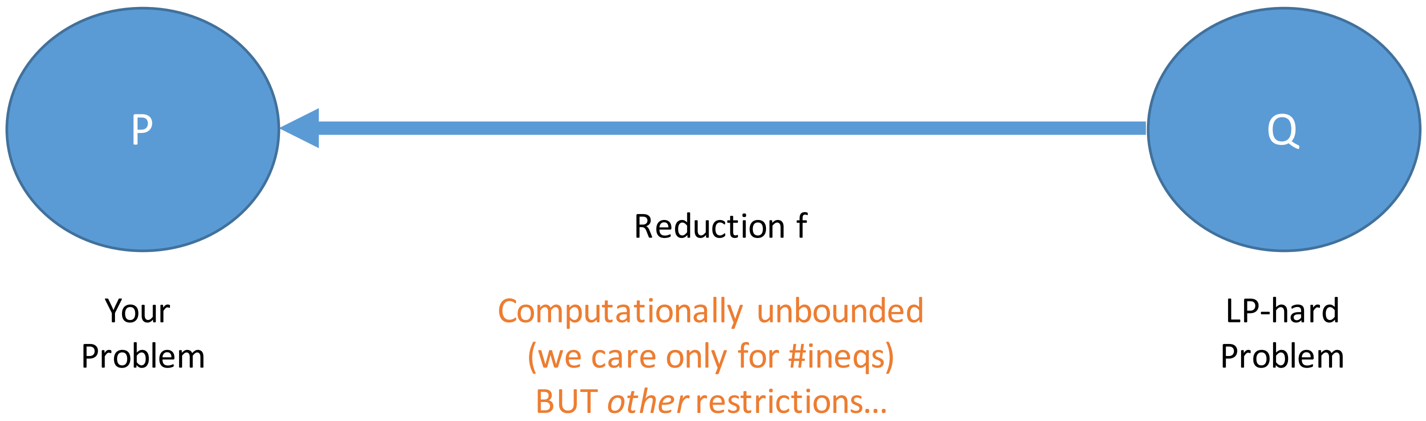


Intuition. f embeds instances of Q into P. Now if P would be easy, then together with f, the problem Q would be easy. Contradiction to hardness of Q.

Note. Reduction can be also used in reverse to establish upper bounds on the complexity of Q if P is known to be easy.

Reductions for EFs

Now for EFs. How to show that a problem P does not admit small LPs?



Intuition. f embeds instances of Q into P. However, somehow this has to preserve the structure of being an LP or SDP…

Note. Reduction can be also used in reverse again. This can lead to interesting results.

Reductions for LPs and SDPs

Affine reductions. $P_1 = (S_1, F_1, \kappa_1, \tau_1)$ reduces to $P_2 = (S_2, F_2, \kappa_2, \tau_2)$ via two maps

1. *Rewrite feasible solutions:* $*: S_1 \to S_2$ with $s_1 \mapsto s_1^* \in S_2$
2. *Rewrite objective functions:* $*: F_1 \to F_2$ with $f_1 \mapsto f_1^* \in F_2$

So that the following holds:

$$\kappa_1(f_1) - f_1(s_1) = [\kappa_2(f_1^*) - f_1^*(s_1^*)] \cdot M_1(f_1, s_1) + M_2(f_1, s_1) \quad \text{(completeness)}$$

$$\max f_1^* \leq \tau_2(f_1^*) \quad \text{if} \quad \max f_1 \leq \tau_1(f_1) \quad \text{(soundness)}$$

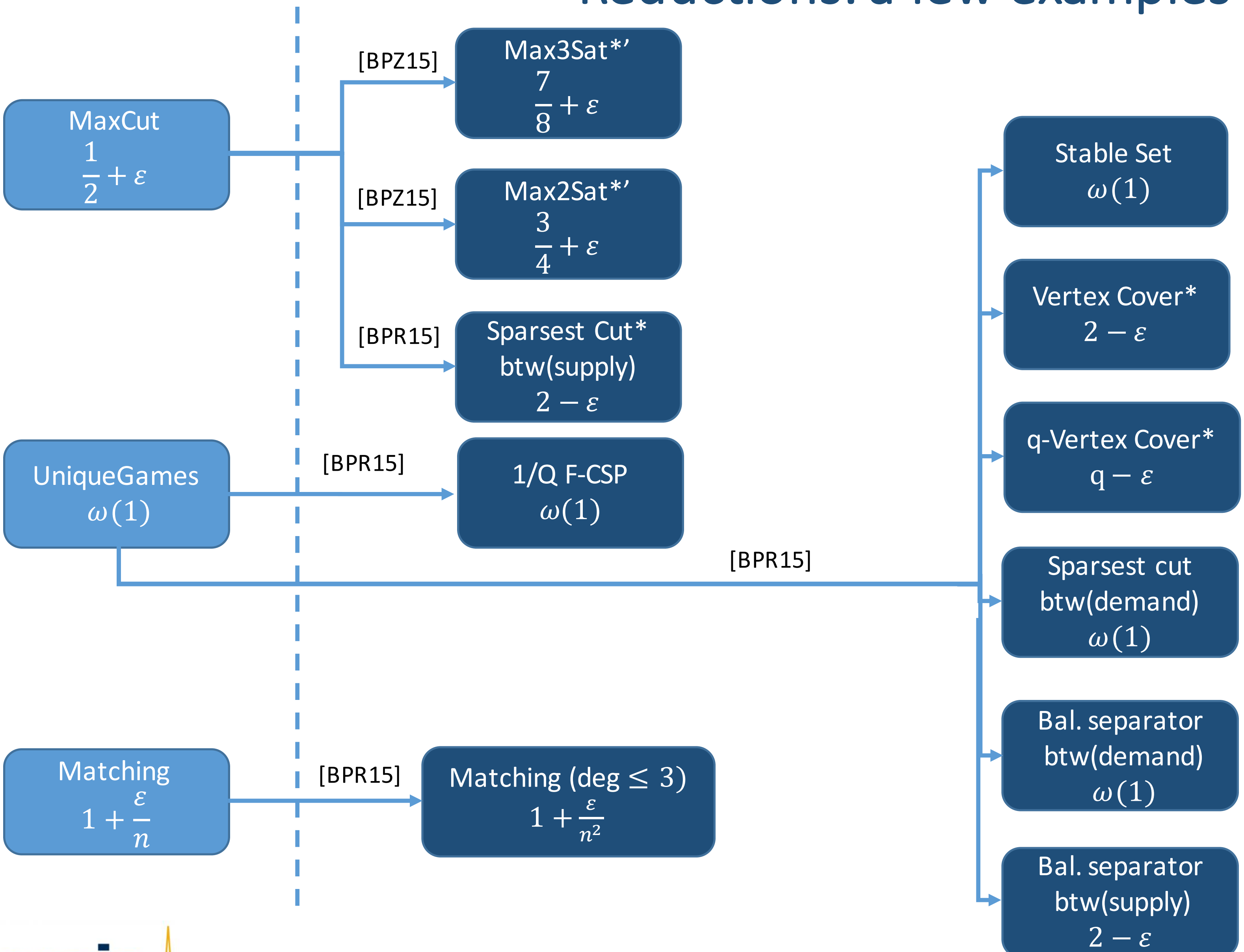
Important. $*$ maps solutions and functions independently of each other.
$M_1, M_2 \in R_+^{r \times d}$ capture low-rank non-affine part of function

Theorem. [Braun, P., Roy 15] If P_1 reduces to P_2, then (essentially)

$$\text{fc}(P_1) \leq rank_{LP/SDP}(M_1) \cdot \text{fc}(P_2) + rank_{LP/SDP}(M_2)$$

Enables certain forms of gap-amplification.

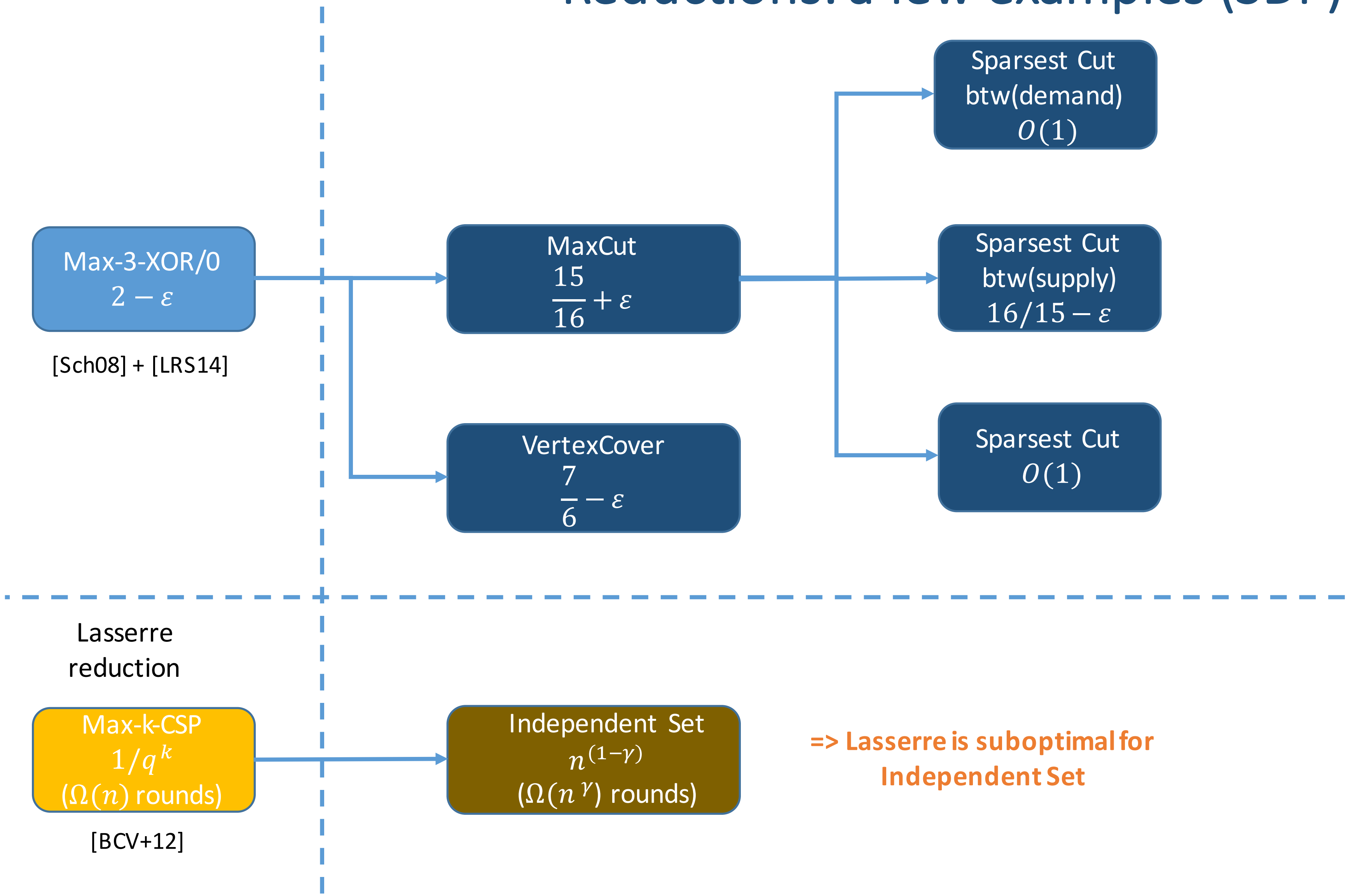
Reductions: a few examples (LP)



* Optimal

‘ Have been also obtained directly

Reductions: a few examples (SDP)



Open Problems

1. (general) SDP formulation complexity of the matching problem
2. Lower bounding techniques for SDP EFs
3. LP-hardness of approximation for complex problems such e.g., TSP
4. Extension complexity over alternative cones:
   1. Hyperbolic programming (in P)
   2. Copositive programming (NP hard)
5. Understanding the difference between hierarchies and general EFs

Thank you!