A "joint+marginal" algorithm for 0/1 optimization

Jean B. Lasserre

LAAS-CNRS and Institute of Mathematics, Toulouse, France

Journées FRO, Paris, March 2010

- Semidefinite Programming
- The "joint+marginal" approach
- Parametric Optimization
- Application to 0/1 optimization
- Some experiments on MAXCUT, *k*-cluster, Knapsack.

- Semidefinite Programming
- The "joint+marginal" approach
- Parametric Optimization
- Application to 0/1 optimization
- Some experiments on MAXCUT, *k*-cluster, Knapsack.

- Semidefinite Programming
- The "joint+marginal" approach
- Parametric Optimization
- Application to 0/1 optimization
- Some experiments on MAXCUT, k-cluster, Knapsack.

- Semidefinite Programming
- The "joint+marginal" approach
- Parametric Optimization
- Application to 0/1 optimization
- Some experiments on MAXCUT, k-cluster, Knapsack.

- Semidefinite Programming
- The "joint+marginal" approach
- Parametric Optimization
- Application to 0/1 optimization
- Some experiments on MAXCUT, *k*-cluster, Knapsack.

The CONVEX optimization problem:

$$\mathbf{P} \quad \rightarrow \quad \min_{x \in \mathbb{R}^n} \{ c' x \mid \sum_{i=1}^{n} A_i x_i \succeq b \},$$

is called a semidefinite program with DUAL:

$$\mathbf{P}^* \quad \rightarrow \quad \max_{Y \in \mathcal{S}_m} \{ \langle b, Y \rangle \mid Y \succeq 0; \langle A_i, Y \rangle = c_i, \quad i = 1, \ldots, n \}$$

- $c \in \mathbb{R}^n$ and $b, A_i, Y \in \mathcal{S}_m$ ($m \times m$ symmetric matrices)
- $Y \succeq 0$ means Y semidefinite positive; $\langle A, B \rangle = \mathrm{trace}\,(AB)$.

Semidefinite Programming

The CONVEX optimization problem:

$$\mathbf{P} \quad \to \quad \min_{x \in \mathbb{R}^n} \{ c' x \mid \sum_{i=1}^n A_i x_i \succeq b \},$$

is called a semidefinite program with DUAL:

$$\mathbf{P}^* \quad \to \quad \max_{Y \in \mathcal{S}_m} \{ \langle b, Y \rangle \mid Y \succeq 0; \ \langle A_i, Y \rangle = c_i, \quad i = 1, \ldots, n \}$$

- $c \in \mathbb{R}^n$ and $b, A_i, Y \in \mathcal{S}_m$ ($m \times m$ symmetric matrices)
- $Y \succeq 0$ means Y semidefinite positive; $\langle A, B \rangle = \mathrm{trace}\,(AB)$.

P and its dual **P*** are **convex** problems that are solvable in polynomial time to arbitrary precision $\epsilon > 0$.
= generalization to the convex cone $\mathcal{S}_m^+$ ($X \succeq 0$) of Linear Programming on the convex polyhedral cone $\mathbb{R}_+^m$ ($x \geq 0$).

Several academic SDP software packages exist, (e.g. MATLAB "LMI toolbox", SeduMi, SDPT3, ...). However, so far, size limitation is more severe than for LP software packages.
Pioneer contributions by A. Nemirovsky, Y. Nesterov, N.Z. Shor, B.D. Yudin,...

P and its dual **P*** are **convex** problems that are solvable in polynomial time to arbitrary precision $\epsilon > 0$.
= generalization to the convex cone $\mathcal{S}_m^+$ ($X \succeq 0$) of Linear Programming on the convex polyhedral cone $\mathbb{R}_+^m$ ($x \geq 0$).

Several academic SDP software packages exist, (e.g. MATLAB "LMI toolbox", SeduMi, SDPT3, ...). However, so far, size limitation is more severe than for LP software packages.
Pioneer contributions by A. Nemirovsky, Y. Nesterov, N.Z. Shor, B.D. Yudin,...

P and its dual **P*** are **convex** problems that are solvable in polynomial time to arbitrary precision $\epsilon > 0$.
= generalization to the convex cone $\mathcal{S}_m^+$ $(X \succeq 0)$ of Linear Programming on the convex polyhedral cone $\mathbb{R}_+^m$ $(x \geq 0)$.

Several academic SDP software packages exist, (e.g. MATLAB “LMI toolbox”, SeduMi, SDPT3, ...). However, so far, size limitation is more severe than for LP software packages.
Pioneer contributions by A. Nemirovsky, Y. Nesterov, N.Z. Shor, B.D. Yudin,...

Consider the 0/1 polynomial optimization problem

$$\mathbf{P}: \quad f^* = \min \{ \mathbf{f}(\mathbf{x}) \, : \, \mathbf{x} \in \mathbf{K}; \, \mathbf{x} \in \{0,1\}^n \}$$

where $\mathbf{K} \subset \mathbb{R}^n$ is the basic semi-algebraic set

$$\mathbf{K} := \{ \mathbf{x} \in \mathbb{R}^n \, : \, g_j(\mathbf{x}) \geq 0, \, j = 1, \ldots, m \}$$

for some polynomials $(\mathbf{f}, g_j) \subset \mathbb{R}[\mathbf{x}]$.

Semidefinite-relaxations

One may define a hierarchy of semidefinite relaxations with optimal value f_k^* such that $f_k^* \uparrow f^*$ as $k \to \infty$. In fact finite convergence takes place and $f_k^* = f^*$ for every $k \geq k_0$.

Moreover, ... practice seems to reveal that in general, convergence is fast ..

Consider the 0/1 polynomial optimization problem

$$\mathbf{P}: \quad f^* = \min \{ \mathbf{f}(\mathbf{x}) \, : \, \mathbf{x} \in \mathbf{K}; \; \mathbf{x} \in \{0,1\}^n \}$$

where $\mathbf{K} \subset \mathbb{R}^n$ is the basic semi-algebraic set

$$\mathbf{K} := \{ \mathbf{x} \in \mathbb{R}^n \, : \, g_j(\mathbf{x}) \geq 0, \; j = 1, \ldots, m \}$$

for some polynomials $(\mathbf{f}, g_j) \subset \mathbb{R}[\mathbf{x}]$.

Semidefinite-relaxations

One may define a hierarchy of semidefinite relaxations with optimal value f_k^* such that $f_k^* \uparrow f^*$ as $k \to \infty$. In fact finite convergence takes place and $f_k^* = f^*$ for every $k \geq k_0$.

Moreover, ... practice seems to reveal that in general, convergence is fast ..

However .....

The size of the k-th semidefinite relaxation grows like $O(n^k)$ and in view of the present status of SDP-solvers, only the first (sometimes the second) relaxation can be implemented, providing only a lower bound f_k^* on f^*!

So .... an important issue is:

How can we use the result of the k-th semidefinite relaxation to help obtain a (good) feasible solution for problem **P**?

Example: After solving the first semidefinite relaxation ($k = 1$), the randomized rounding procedure for MAXCUT provides an approximate solution with guaranteed performance!

However .....

The size of the k-th semidefinite relaxation grows like $O(n^k)$ and in view of the present status of SDP-solvers, only the first (sometimes the second) relaxation can be implemented, providing only a lower bound f^*_k on f^*!

So .... an important issue is:

How can we use the result of the k-th semidefinite relaxation to help obtain a (good) feasible solution for problem **P**?

Example: After solving the first semidefinite relaxation ($k = 1$), the randomized rounding procedure for MAXCUT provides an approximate solution with guaranteed performance!

However .....

The size of the k-th semidefinite relaxation grows like $O(n^k)$ and in view of the present status of SDP-solvers, only the first (sometimes the second) relaxation can be implemented, providing only a lower bound f_k^* on f^*!

So .... an important issue is:

How can we use the result of the k-th semidefinite relaxation to help obtain a (good) feasible solution for problem **P**?

Example: After solving the first semidefinite relaxation ($k = 1$), the randomized rounding procedure for MAXCUT provides an approximate solution with guaranteed performance!

The underlying idea

Let $\mathbf{Y} := \{0, 1\}$ and let $y \in \mathbf{Y}$, fixed:

Consider the y-parametric optimization problem

$$J(y) = \min_{\mathbf{x}} \{ \mathbf{f}(\mathbf{x}) \; : \; \mathbf{x} \in \mathbf{K}; \; \mathbf{x} \in \{0, 1\}^n; \; x_1 = y \}$$

i.e., problem **P** where the variable x_1 is fixed at the value y

Of course ...

$$f^* = \min_{y} \{ J(y) \; : \; y \in \mathbf{Y} \}$$

The underlying idea

Let $\mathbf{Y} := \{0, 1\}$ and let $y \in \mathbf{Y}$, fixed:

Consider the y-parametric optimization problem

$$J(y) = \min_{\mathbf{x}} \{ \mathbf{f}(\mathbf{x}) \; : \; \mathbf{x} \in \mathbf{K}; \; \mathbf{x} \in \{0, 1\}^n; \; x_1 = y \}$$

i.e., problem **P** where the variable x_1 is fixed at the value y

Of course ...

$$f^* = \min_{y} \{ J(y) \; : \; y \in \mathbf{Y} \}$$

Suppose that one has an approximation $J_k : \mathbf{Y} \to \mathbb{R}$ such that $J_k(y) \to \rho(y)$ as $k \to \infty$.

Then .... a (likely) reasonable strategy is:

- Select $x_1 := 0$ if $J_k(0) < J_k(1)$ and select $x_1 := 1$ otherwise!
- repeat with the $(n-1)$-variable 0/1 problem:

$$\mathbf{P}(x_1): \quad \min\{\,\mathbf{f}(\mathbf{x})\;:\;\mathbf{x}\in\mathbf{K};\;\mathbf{x}\in\{0,1\}^n;\;x_1=x_1\}$$

and its associated y-parametric optimization problem:

$$J(y) = \min_{\mathbf{x}}\{\,\mathbf{f}(\mathbf{x})\;:\;\mathbf{x}\in\mathbf{K};\;\mathbf{x}\in\{0,1\}^n;\;x_1=x_1;\;x_2=y\}$$

i.e., problem $\mathbf{P}$ where the variable x_1 is fixed at the value x_1, and the variable x_2 is fixed at the value $y \in \mathbf{Y}$.

- etc. until one obtains $\mathbf{x} \in \{0,1\}^n$.

Suppose that one has an approximation $J_k : \mathbf{Y} \to \mathbb{R}$ such that $J_k(y) \to \rho(y)$ as $k \to \infty$.

Then .... a (likely) reasonable strategy is:

- Select $x_1 := 0$ if $J_k(0) < J_k(1)$ and select $x_1 := 1$ otherwise!
- repeat with the $(n-1)$-variable 0/1 problem:

$$\mathbf{P}(x_1): \quad \min\{\,\mathbf{f}(\mathbf{x}) \;:\; \mathbf{x} \in \mathbf{K};\; \mathbf{x} \in \{0,1\}^n;\; x_1 = x_1\,\}$$

and its associated y-parametric optimization problem:

$$J(y) = \min_{\mathbf{x}} \{\,\mathbf{f}(\mathbf{x}) \;:\; \mathbf{x} \in \mathbf{K};\; \mathbf{x} \in \{0,1\}^n;\; x_1 = x_1;\; x_2 = y\}$$

i.e., problem **P** where the variable x_1 is fixed at the value x_1, and the variable x_2 is fixed at the value $y \in \mathbf{Y}$.

- etc. until one obtains $\mathbf{x} \in \{0,1\}^n$.

Features

- For problems where feasibility is easy to determine (e.g., MAXCUT, k-cluster, $0/1$-knapsack, ...), one ends up with a feasible $\mathbf{x} \in \{0,1\}^n$.
- To compute $J_k(y)$ one does NOT need to solve 2 semidefinite relaxations to get $J_k(0)$ AND $J_k(1)$ as in a Branch and Bound procedure. It suffices to compute the k-th semidefinite relaxation associated with **P**, with k additional linear constraints!
- An optimal solution of the dual provides us with the function $y \mapsto J_k(y)$, a linear polynomial $\lambda_0 + \lambda_1 y$.
- and so one selects $x_1 = 0$ if $\lambda_1 > 0$ and $x_1 = 1$ otherwise.

Features

- For problems where feasibility is easy to determine (e.g., MAXCUT, k-cluster, $0/1$-knapsack, ...), one ends up with a feasible $\mathbf{x} \in \{0, 1\}^n$.
- To compute $J_k(y)$ one does NOT need to solve 2 semidefinite relaxations to get $J_k(0)$ AND $J_k(1)$ as in a Branch and Bound procedure. It suffices to compute the k-th semidefinite relaxation associated with **P**, with k additional linear constraints!
- An optimal solution of the dual provides us with the function $y \mapsto J_k(y)$, a linear polynomial $\lambda_0 + \lambda_1 y$.
- and so one selects $x_1 = 0$ if $\lambda_1 > 0$ and $x_1 = 1$ otherwise.

Features

- For problems where feasibility is easy to determine (e.g., MAXCUT, k-cluster, $0/1$-knapsack, ...), one ends up with a feasible $\mathbf{x} \in \{0, 1\}^n$.
- To compute $J_k(y)$ one does NOT need to solve 2 semidefinite relaxations to get $J_k(0)$ AND $J_k(1)$ as in a Branch and Bound procedure. It suffices to compute the k-th semidefinite relaxation associated with **P**, with k additional linear constraints!
- An optimal solution of the dual provides us with the function $y \mapsto J_k(y)$, a linear polynomial $\lambda_0 + \lambda_1 y$.
- and so one selects $x_1 = 0$ if $\lambda_1 > 0$ and $x_1 = 1$ otherwise.

Features

- For problems where feasibility is easy to determine (e.g., MAXCUT, k-cluster, $0/1$-knapsack, ...), one ends up with a feasible $\mathbf{x} \in \{0,1\}^n$.
- To compute $J_k(y)$ one does NOT need to solve 2 semidefinite relaxations to get $J_k(0)$ AND $J_k(1)$ as in a Branch and Bound procedure. It suffices to compute the k-th semidefinite relaxation associated with **P**, with k additional linear constraints!
- An optimal solution of the dual provides us with the function $y \mapsto J_k(y)$, a linear polynomial $\lambda_0 + \lambda_1 y$.
- and so one selects $x_1 = 0$ if $\lambda_1 > 0$ and $x_1 = 1$ otherwise.

- The same approach can be done with a block of s parameters $(y_1, \ldots, y_s) \in \mathbf{Y} := \{0,1\}^s$. To compute $J_k(y_1, \ldots, y_s)$, one only needs to solve ONE k-th semidefinite relaxation with $O(s^{2k})$ additional linear constraints instead of solving 2^s semidefinite relaxations!
- The function $(y_1, \ldots y_s) \mapsto J_k(y_1, \ldots, y_s)$ is a (square free) polynomial of degree s.

$$J_k(y_1, \ldots y_s) = \lambda_0 + \sum_{i=1}^{s} \lambda_i \, y_i + \sum_{1 \leq i < j \leq s} \lambda_{ij} \, y_i y_j + \cdots$$

- Select $(x_1, \ldots, x_s) \in \{0,1\}^s$ that minimizes J_k on $\mathbf{Y}$ by inspection of the corresponding 2^s values of J_k.
- Repeat with the $(n-s)$-variable problem $\mathbf{P}(x_1, , \ldots, x_s)$:

$$\min_{\mathbf{x}} \{ \, \mathbf{f}(\mathbf{x}) \; : \; \mathbf{x} \in \mathbf{K}; \; \mathbf{x} \in \{0,1\}^n; \; x_k = x_k, \; k = 1, \ldots, s \}$$

and associated $(y_{s+1}, \ldots, y_{2s})$-parametric problem, etc.

- The same approach can be done with a block of s parameters $(y_1, \ldots, y_s) \in \mathbf{Y} := \{0,1\}^s$. To compute $J_k(y_1, \ldots, y_s)$, one only needs to solve ONE k-th semidefinite relaxation with $O(s^{2k})$ additional linear constraints instead of solving 2^s semidefinite relaxations!
- The function $(y_1, \ldots y_s) \mapsto J_k(y_1, \ldots, y_s)$ is a (square free) polynomial of degree s.

$$J_k(y_1, \ldots y_s) = \lambda_0 + \sum_{i=1}^{s} \lambda_i\, y_i + \sum_{1 \leq i < j \leq s} \lambda_{ij}\, y_i y_j + \cdots$$

- Select $(x_1, \ldots, x_s) \in \{0,1\}^s$ that minimizes J_k on $\mathbf{Y}$ by inspection of the corresponding 2^s values of J_k.
- Repeat with the $(n-s)$-variable problem $\mathbf{P}(x_1, , \ldots, x_s)$:

$$\min_{\mathbf{x}} \{ \mathbf{f}(\mathbf{x}) \; : \; \mathbf{x} \in \mathbf{K};\; \mathbf{x} \in \{0,1\}^n;\; x_k = x_k,\; k = 1, \ldots, s\}$$

and associated $(y_{s+1}, \ldots, y_{2s})$-parametric problem, etc.

- The same approach can be done with a block of s parameters $(y_1,\ldots,y_s) \in \mathbf{Y} := \{0,1\}^s$. To compute $J_k(y_1,\ldots,y_s)$, one only needs to solve ONE k-th semidefinite relaxation with $O(s^{2k})$ additional linear constraints instead of solving 2^s semidefinite relaxations!
- The function $(y_1,\ldots y_s) \mapsto J_k(y_1,\ldots,y_s)$ is a (square free) polynomial of degree s.

$$J_k(y_1,\ldots y_s) = \lambda_0 + \sum_{i=1}^{s} \lambda_i\, y_i + \sum_{1\leq i<j\leq s} \lambda_{ij}\, y_i y_j + \cdots$$

- Select $(x_1,\ldots,x_s) \in \{0,1\}^s$ that minimizes J_k on $\mathbf{Y}$ by inspection of the corresponding 2^s values of J_k.
- Repeat with the $(n-s)$-variable problem $\mathbf{P}(x_1,,\ldots,x_s)$:

$$\min_{\mathbf{x}}\{\,\mathbf{f}(\mathbf{x}) \;:\; \mathbf{x}\in\mathbf{K};\ \mathbf{x}\in\{0,1\}^n;\ x_k = x_k,\ k=1,\ldots,s\}$$

and associated $(y_{s+1},\ldots,y_{2s})$-parametric problem, etc.

Parametric Optimization

Let $\mathbf{Y} \subset \mathbb{R}^p$ be a compact set, called the parameter set.

Let $\mathbf{K} \subset \mathbb{R}^n \times \mathbb{R}^p$ be the set:

$$\mathbf{K} := \{ (\mathbf{x}, \mathbf{y}) \,:\, \mathbf{y} \in \mathbf{Y}; \quad h_j(\mathbf{x}, \mathbf{y}) \geq 0, \quad j = 1, \ldots, m \},$$

for some continuous functions $h_j : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$.

Consider the following optimization problem:

$$J(\mathbf{y}) := \inf_{\mathbf{x}} \{ f(\mathbf{x}, \mathbf{y}) \,:\, x \in \mathbf{K}_{\mathbf{y}} \},$$

where for each $\mathbf{y} \in \mathbf{Y}$, the $\mathbf{K}_{\mathbf{y}} \subset \mathbb{R}^n$ is defined by:

$$\mathbf{K}_{\mathbf{y}} := \{ \mathbf{x} \in \mathbb{R}^n \,:\, (\mathbf{x}, \mathbf{y}) \in \mathbf{K} \}$$

Parametric optimization is concerned with:

- the global optimal value function $\mathbf{y} \mapsto J(\mathbf{y})$, and
- the global minimizer set function $\mathbf{y} \mapsto \mathbf{x}_i^*(\mathbf{y})$
- the optimal dual multiplier set function $\mathbf{y} \mapsto \lambda_j^*(\mathbf{y})$ associated with the constraint $h_j(\mathbf{x}, \mathbf{y}) \geq 0$.

In general, getting full information is impossible, and one is satisfied with local information (e.g. sensitivity analysis) around some (even local) minimizer $\mathbf{x}^*(\mathbf{y}) \in \mathbf{K}_{\mathbf{y}}$, $\mathbf{y} \in \mathbf{Y}$. (See e.g. the book by Bonnans and Shapiro.)

However ...

For polynomial optimization much more is possible!

Parametric optimization is concerned with:

- the global optimal value function $\mathbf{y} \mapsto J(\mathbf{y})$, and
- the global minimizer set function $\mathbf{y} \mapsto \mathbf{x}_i^*(\mathbf{y})$
- the optimal dual multiplier set function $\mathbf{y} \mapsto \lambda_j^*(\mathbf{y})$ associated with the constraint $h_j(\mathbf{x}, \mathbf{y}) \geq 0$.

In general, getting full information is impossible, and one is satisfied with local information (e.g. sensitivity analysis) around some (even local) minimizer $\mathbf{x}^*(\mathbf{y}) \in \mathbf{K_y}$, $\mathbf{y} \in \mathbf{Y}$. (See e.g. the book by Bonnans and Shapiro.)

However ...

For polynomial optimization much more is possible!

Parametric optimization is concerned with:

- the global optimal value function $\mathbf{y} \mapsto J(\mathbf{y})$, and
- the global minimizer set function $\mathbf{y} \mapsto \mathbf{x}_i^*(\mathbf{y})$
- the optimal dual multiplier set function $\mathbf{y} \mapsto \lambda_j^*(\mathbf{y})$ associated with the constraint $h_j(\mathbf{x}, \mathbf{y}) \geq 0$.

In general, getting full information is impossible, and one is satisfied with local information (e.g. sensitivity analysis) around some (even local) minimizer $\mathbf{x}^*(\mathbf{y}) \in \mathbf{K}_{\mathbf{y}}$, $\mathbf{y} \in \mathbf{Y}$. (See e.g. the book by Bonnans and Shapiro.)

However ...

For polynomial optimization much more is possible!

The "joint+marginal" approach

Let φ be a Borel probability measure on $\mathbf{Y}$, with a positive density with respect to the Lebesgue measure on the smallest affine variety that contains $\mathbf{Y}$. For instance,

$$\varphi(B) := \left(\int_{\mathbf{Y}} d\mathbf{y}\right)^{-1} \int_B d\mathbf{y}, \qquad \forall B \in \mathcal{B}(\mathbf{Y}),$$

is uniformly distributed on $\mathbf{Y}$.

For a discrete set of parameters $\mathbf{Y}$ (finite or countable) take for φ a discrete probability measure on $\mathbf{Y}$ with strictly positive weight at each point of the support.

Sometimes, e.g. in the context of optimization with data uncertainty, φ is already specified.

A related infinite-dimensional linear program:

Consider the infinite-dimensional LP:

$$\mathbf{P}: \quad \rho := \inf_{\mu \in \mathbf{M}(\mathbf{K})} \left\{ \int_{\mathbf{K}} f \, d\mu \; : \; \pi\mu = \varphi \right\}$$

where: $\mathbf{M}(\mathbf{K})$ is the of Borel probability measures on $\mathbf{K}$, and

$\pi : \mathbf{M}(\mathbf{K}) \to \mathbf{M}(\mathbf{Y})$ is the projection (or, marginal) on $\mathbf{Y}$.

Whence the name "joint+marginal"-approach since:

- μ is a joint distribution on the variables $\mathbf{x}$ AND the parameters $\mathbf{y}$.
- φ is the marginal of μ on $\mathbf{Y}$ (fixed, as a constraint on μ).

A related infinite-dimensional linear program:

Consider the infinite-dimensional LP:

$$\mathbf{P}: \quad \rho := \inf_{\mu \in \mathbf{M}(\mathbf{K})} \left\{ \int_{\mathbf{K}} f\, d\mu \ : \ \pi\mu = \varphi \right\}$$

where: $\mathbf{M}(\mathbf{K})$ is the of Borel probability measures on $\mathbf{K}$, and

$\pi : \mathbf{M}(\mathbf{K}) \to \mathbf{M}(\mathbf{Y})$ is the projection (or, marginal) on $\mathbf{Y}$.

Whence the name "joint+marginal"-approach since:

- μ is a joint distribution on the variables $\mathbf{x}$ AND the parameters $\mathbf{y}$.
- φ is the marginal of μ on $\mathbf{Y}$ (fixed, as a constraint on μ).

The dual $\mathbf{P}^*$ is the infinite-dimensional LP:

$$\mathbf{P}^*: \quad \rho^* := \sup_{g \in C(\mathbf{Y})} \int_{\mathbf{Y}} g(\mathbf{y})\, d\varphi(\mathbf{y})$$
$$f(\mathbf{x},\mathbf{y}) - g(\mathbf{y}) \geq 0 \quad \forall (\mathbf{x},\mathbf{y}) \in \mathbf{K}.$$

where $C(\mathbf{Y})$ is the set of continuous functions on $\mathbf{Y}$.

In other words, among the continuous functions g on $\mathbf{Y}$ such that:

$$f(\mathbf{x},\mathbf{y}) \geq g(\mathbf{y}) \qquad \forall \mathbf{x} \in \mathbf{K}_{\mathbf{y}},$$

one searches for the one that maximizes $\int_{\mathbf{Y}} g d\varphi$.

The dual $\mathbf{P}^*$ is the infinite-dimensional LP:

$$\mathbf{P}^* : \quad \rho^* := \sup_{g \in C(\mathbf{Y})} \int_{\mathbf{Y}} g(\mathbf{y})\, d\varphi(\mathbf{y})$$
$$f(\mathbf{x}, \mathbf{y}) - g(\mathbf{y}) \geq 0 \quad \forall (\mathbf{x}, \mathbf{y}) \in \mathbf{K}.$$

where $C(\mathbf{Y})$ is the set of continuous functions on $\mathbf{Y}$.

In other words, among the continuous functions g on $\mathbf{Y}$ such that:

$$f(\mathbf{x}, \mathbf{y}) \geq g(\mathbf{y}) \qquad \forall \mathbf{x} \in \mathbf{K}_{\mathbf{y}},$$

one searches for the one that maximizes $\int_{\mathbf{Y}} g d\varphi$.

Why those LPs?

We assume that $\mathbf{K}$ is compact.

As we shall see ....

Any optimal solution μ^* of the primal $\mathbf{P}$ encodes *all* information on the optimal solutions $\mathbf{x}^*(\mathbf{y})$ of $\mathbf{P}_\mathbf{y}$.

Similarly ....

There is no duality gap $\rho = \rho^*$ and so, in particular, the optimal value function $\mathbf{y} \mapsto J(\mathbf{y})$ of $\mathbf{P}_\mathbf{y}$ can be nicely approximated by polynomials.

Why those LPs?

We assume that $\mathbf{K}$ is compact.

As we shall see ....

Any optimal solution μ^* of the primal $\mathbf{P}$ encodes *all* information on the optimal solutions $\mathbf{x}^*(\mathbf{y})$ of $\mathbf{P}_{\mathbf{y}}$.

Similarly ....

There is no duality gap $\rho = \rho^*$ and so, in particular, the optimal value function $\mathbf{y} \mapsto J(\mathbf{y})$ of $\mathbf{P}_{\mathbf{y}}$ can be nicely approximated by polynomials.

Theorem (The primal side ...)

Assume that $\mathbf{K}$ *is compact and* $\mathbf{K}_{\mathbf{y}} \neq \emptyset$ *for every* $\mathbf{y} \in \mathbf{Y}$. *Let*

$\mathbf{X}^*_{\mathbf{y}} := \{\mathbf{x} \in \mathbb{R}^n \,:\, f(\mathbf{x}, \mathbf{y}) = J(\mathbf{y})\}$, $\mathbf{y} \in \mathbf{Y}$. *Then:*

(a) $\rho = \int_{\mathbf{Y}} J(\mathbf{y})\, d\varphi(\mathbf{y})$ *and* $\mathbf{P}$ *has an optimal solution.*

(b) *For every optimal solution* μ^* *of* $\mathbf{P}$, *and for* φ*-almost all* $\mathbf{y} \in \mathbf{Y}$, *there is a probability measure* $\psi^*(d\mathbf{x} \,|\, \mathbf{y})$ *on* $\mathbb{R}^n$, *concentrated on* $\mathbf{X}^*_{\mathbf{y}}$, *such that:*

$$\mu^*(C \times B) = \int_B \psi^*(C \,|\, \mathbf{y})\, d\varphi(\mathbf{y}), \qquad \forall B \in \mathcal{B}(\mathbf{Y}),\; C \in \mathcal{B}(\mathbb{R}^n).$$

continued ...

(c) Assume that for φ-almost all $\mathbf{y} \in \mathbf{Y}$, the set of minimizers $\mathbf{X}^*_{\mathbf{y}}$ is the singleton $\{\mathbf{x}^*(\mathbf{y})\}$ for some $\mathbf{x}^*(\mathbf{y}) \in \mathbf{K_y}$. Then there is a measurable mapping $g : \mathbf{Y} \to \mathbf{K_y}$ such that

$$g(\mathbf{y}) = \mathbf{x}^*(\mathbf{y}) \text{ for every } \mathbf{y} \in \mathbf{Y}; \quad \rho = \int_{\mathbf{Y}} f(g(\mathbf{y}), \mathbf{y})\, d\varphi(\mathbf{y}),$$

and for every $\alpha \in \mathbb{N}^n$, and $\beta \in \mathbb{N}^p$:

$$\int_{\mathbf{K}} \mathbf{x}^\alpha \mathbf{y}^\beta\, d\mu^*(\mathbf{x}, \mathbf{y}) = \int_{\mathbf{Y}} \mathbf{y}^\beta\, g(\mathbf{y})^\alpha\, d\varphi(\mathbf{y}).$$

Theorem (The dual side ...)

(a) There is no duality gap, i.e.,

$$\rho = \rho^* = \int_{\mathbf{Y}} J(\mathbf{y})\, d\varphi(\mathbf{y}),$$

(b) One may use polynomials of $\mathbb{R}[\mathbf{y}]$ to approximate ρ^*.

(c) Let $(p_i) \subset \mathbb{R}[\mathbf{y}]$ be any maximizing sequence. Then:

L_1-norm convergence:

$$\textit{as } i \to \infty, \qquad \int_{\mathbf{Y}} |J(\mathbf{y}) - p_i(\mathbf{y})|\, d\varphi(\mathbf{y}) \to 0$$

φ-almost sure convergence: *Let* $\tilde{p}_i := \max_{k=0,..,i} p_i$. *Then*

$$\textit{as } i \to \infty, \qquad \tilde{p}_i \to J \qquad \varphi\textit{-almost surely in } \mathbf{Y}$$

Polynomial Parametric Optimization

In general, P and P* are intractable!

However .... when:

- **Y** and **K**, are basic semi-algebraic sets, and:
- either one already knows the moments of φ, or **Y** is simple enough (e.g. a box, a simplex, a hyper-sphere) so that they can be computed.

.... then one can approximate the optimal value ρ of P, and:

- The optimal value mapping $\mathbf{y} \mapsto J(\mathbf{y})$
- The global minimizer mapping $\mathbf{y} \mapsto \mathbf{x}^*(\mathbf{y})$,

... via the hierarchy of semidefinite relaxations

adapted from the moment-s.o.s. approach in polynomial optimization.

Polynomial Parametric Optimization

In general, P and P* are intractable!

However .... when:

- **Y** and **K**, are basic semi-algebraic sets, and:
- either one already knows the moments of φ, or **Y** is simple enough (e.g. a box, a simplex, a hyper-sphere) so that they can be computed.

.... then one can approximate the optimal value ρ of **P**, and:

- The optimal value mapping $\mathbf{y} \mapsto J(\mathbf{y})$
- The global minimizer mapping $\mathbf{y} \mapsto \mathbf{x}^*(\mathbf{y})$,

... via the hierarchy of semidefinite relaxations

adapted from the moment-s.o.s. approach in polynomial optimization.

In general, P and P* are intractable!

However .... when:

- **Y** and **K**, are basic semi-algebraic sets, and:
- either one already knows the moments of φ, or **Y** is simple enough (e.g. a box, a simplex, a hyper-sphere) so that they can be computed.

.... then one can approximate the optimal value ρ of **P**, and:

- The optimal value mapping $\mathbf{y} \mapsto J(\mathbf{y})$
- The global minimizer mapping $\mathbf{y} \mapsto \mathbf{x}^*(\mathbf{y})$,

... via the hierarchy of semidefinite relaxations

adapted from the moment-s.o.s. approach in polynomial optimization.

More details in:

The "joint+marginal" approach for parametric optimization

SIAM J. Optim. **20** (2010).

A "joint+marginal" algorithm for 0/1 optimization

With $\mathbf{K} \subset \mathbb{R}^n$ being the basic semi-algebraic set

$$\mathbf{K} := \{\mathbf{x} \in \mathbb{R}^n \,:\, g_j(\mathbf{x}) \geq 0, \; j = 1, \ldots, m\}$$

Consider the 0/1 polynomial optimization problem

$$\mathbf{P}: \quad f^* = \min \{ \mathbf{f}(\mathbf{x}) \,:\, \mathbf{x} \in \mathbf{K};\; \mathbf{x} \in \{0,1\}^n \}$$

and its associated y-parametric optimization problem:

$$\rho(y) = \min_{\mathbf{x}} \{ \mathbf{f}(\mathbf{x}) \,:\, \mathbf{x} \in \mathbf{K};\; \mathbf{x} \in \{0,1\}^n;\; x_1 = y \}$$

The moment-s.o.s. approach

Let $\mathbb{N}^n_i := \{\alpha \in \mathbb{N}^n : \sum_j \alpha_j \leq i\}$.

With a sequence $\mathbf{z} = (z_\alpha)$, indexed in the canonical basis (x^α) of $\mathbb{R}[x]$, let $L_\mathbf{z} : \mathbb{R}[x] \to \mathbb{R}$ be the linear mapping:

$$f \left(= \sum_\alpha f_\alpha x^\alpha\right) \mapsto L_\mathbf{z}(f) := \sum_\alpha f_\alpha z_\alpha, \qquad f \in \mathbb{R}[x].$$

The moment matrix $\mathbf{M}_i(\mathbf{z})$

associated with a sequence $\mathbf{z} = (z_\alpha)$, has its rows and columns indexed in the canonical basis (x^α), and with entries.

$$\mathbf{M}_i(\mathbf{z})(\alpha, \beta) = L_\mathbf{z}(x^\alpha x^\beta) = z_{\alpha+\beta},$$

for every $\alpha, \beta \in \mathbb{N}^p_i$

Let q be the polynomial $x \mapsto q(x) := \sum_u q_u \, x^u$.

The localizing matrix $\mathbf{M}_i(q, \mathbf{z})$ associated with

$q \in \mathbb{R}[x]$ and a sequence $\mathbf{z} = (z_\alpha)$, has its rows and columns indexed in the canonical basis (x^α), and with entries.

$$\mathbf{M}_i(q\,\mathbf{z})(\alpha, \beta) \,=\, L_{\mathbf{z}}(q(x)x^\alpha \, x^\beta) \,=\, \sum_{u \in \mathbb{N}^n} q_u \, z_{\alpha+\beta+u}$$

for every $\alpha, \beta \in \mathbb{N}_i^n$.

Primal semidefinite relaxations:

Let $v_j := \lceil (\deg g_j)/2 \rceil$ for every $j = 1, \ldots, m$ and let $i_0 := \max[\lceil (\deg f)/2 \rceil, \max_j v_j]$.

For $k \geq i_0$, consider the semidefinite program:

$$\begin{array}{rll} \rho_k = & \inf\limits_{\mathbf{z}} & L_{\mathbf{z}}(f) \\ & \text{s.t.} & \mathbf{M}_k(\mathbf{z}) \succeq 0 \\ & & \mathbf{M}_{k-v_j}(g_j\,\mathbf{z}) \succeq 0, \quad j = 1, \ldots, m \\ & & L_{\mathbf{z}}(x^\alpha) = L_{\mathbf{z}}(x^{1_{\alpha>0}}), \qquad \forall\, |\alpha| \leq 2k \\ & & L_{\mathbf{z}}(1) = 1 \\ & & L_{\mathbf{z}}(x_1) = 1/2 \end{array}$$

where $1_{\alpha>0} = (1_{\alpha_1>0}, \ldots, 1_{\alpha_n>0})$. (Comes from $x_i{}^2 = x_i$, $\forall i$).

$$\rho_{i_0} \leq \cdots \leq \rho_k \leq \cdots \leq \rho.$$

Dual semidefinite relaxation

The dual reads:

$$
\begin{aligned}
\rho_k^* = \quad & \sup_{\lambda,(\sigma_i)} \lambda_0 + \lambda_1/2 \\
\text{s.t.} \quad & f(x) - (\lambda_0 + \lambda_1\, x_1) \, = \, \sigma_0(x) + \sum_{j=1}^m \sigma_j(x)\, g_j(x), \ \forall x \\
& \sigma_j \in \Sigma[x], \quad j = 1, \ldots, m \\
& \deg \sigma_j g_j \leq 2k, \quad j = 1, \ldots, m
\end{aligned}
$$

Set $y \mapsto J_k(y) := \lambda_0^k + \lambda_1^k\, y$ for an optimal solution $(\lambda_0^k, \lambda_1^k, \sigma_j^k)$, and observe that, and observe that

$$\lambda_0^k + \lambda_1^k/2 \, = \, \int_{\mathbf{Y}} J_k(y)\, d\varphi(y) \, = \, \int_{\{0,1\}} J_k(y)\, d\varphi(y).$$

with $\varphi(\{0\}) = 1/2$ and $\varphi(\{1\}) = 1/2$.

Dual semidefinite relaxation

The dual reads:

$$\rho_k^* = \sup_{\lambda,(\sigma_i)} \lambda_0 + \lambda_1/2$$

$$\text{s.t.} \quad f(x) - (\lambda_0 + \lambda_1\, x_1) = \sigma_0(x) + \sum_{j=1}^m \sigma_j(x)\, g_j(x), \; \forall x$$

$$\sigma_j \in \Sigma[x], \quad j = 1, \ldots, m$$

$$\deg \sigma_j g_j \leq 2k, \quad j = 1, \ldots, m$$

Set $y \mapsto J_k(y) := \lambda_0^k + \lambda_1^k\, y$ for an optimal solution $(\lambda_0^k, \lambda_1^k, \sigma_j^k)$, and observe that, and observe that

$$\lambda_0^k + \lambda_1^k/2 = \int_{\mathbf{Y}} J_k(y)\, d\varphi(y) = \int_{\{0,1\}} J_k(y)\, d\varphi(y).$$

with $\varphi(\{0\}) = 1/2$ and $\varphi(\{1\}) = 1/2$.

Theorem

Consider the dual semidefinite relaxations. Then:

(a) $\rho_k^* \uparrow \rho$ *as* $k \to \infty$.

(b) *Let* $(\lambda_0^k, \lambda_1^k, \sigma_j^k)$ *be an optimal solution. Then:*

$$J_k(0) = \lambda_0^k \leq J(0) \quad \text{and} \quad J_k(1) = \lambda_0^k + \lambda_1^k \leq J(1).$$

Moreover, as $k \to \infty$,

$$J_k(0) = \lambda_0^k \to J(0) \quad \text{and} \quad J_k(1) = \lambda_0^k + \lambda_1^k \to J(1).$$

In fact, finite convergence takes place!

Theorem

Consider the dual semidefinite relaxations. Then:

(a) $\rho_k^* \uparrow \rho$ *as* $k \to \infty$.

(b) *Let* $(\lambda_0^k, \lambda_1^k, \sigma_j^k)$ *be an optimal solution. Then:*

$$J_k(0) = \lambda_0^k \leq J(0) \quad \text{and} \quad J_k(1) = \lambda_0^k + \lambda_1^k \leq J(1).$$

Moreover, as $k \to \infty$,

$$J_k(0) = \lambda_0^k \to J(0) \quad \text{and} \quad J_k(1) = \lambda_0^k + \lambda_1^k \to J(1).$$

In fact, finite convergence takes place!

Theorem

Consider the dual semidefinite relaxations. Then:

(a) $\rho_k^* \uparrow \rho$ *as* $k \to \infty$.

(b) *Let* $(\lambda_0^k, \lambda_1^k, \sigma_j^k)$ *be an optimal solution. Then:*

$$J_k(0) = \lambda_0^k \leq J(0) \quad \text{and} \quad J_k(1) = \lambda_0^k + \lambda_1^k \leq J(1).$$

Moreover, as $k \to \infty$,

$$J_k(0) = \lambda_0^k \to J(0) \quad \text{and} \quad J_k(1) = \lambda_0^k + \lambda_1^k \to J(1).$$

In fact, finite convergence takes place!

Hence .. if k is suffciently large,

$$\lambda_1^k > 0 \Rightarrow x_1^* = 0 \quad \text{and} \quad \lambda_1^k < 0 \Rightarrow x_1^* = 1$$

in any optimal solution x^* of **P**!

... which provides a rationale for the following

"joint+marginal" algorithm for 0/1 optimization

While $i < n$ repeat:

- Consider the 0/1 problem $\mathbf{P}(x_1, \ldots, x_{i-1})$ which is **P** where the first $i-1$ components of $\mathbf{x}$ are already fixed.
- Solve the k-th semidefinite relaxation with parameter x_i associated with $\mathbf{P}(x_1, \ldots, x_{i-1})$, and get an optimal solution $(\lambda_0^k, \lambda_1^k, \sigma_j^k)$ of the dual.
- If $\lambda_1^k > 0$ set $x_i = 0$ and $x_i = 1$ otherwise.
- If $\mathbf{P}(x_1, \ldots, x_{i-1}, x_i)$ has a feasible solution select $x_i := x_i$ and $x_i := 1 - x_i$ otherwise.

Hence .. if k is suffciently large,

$$\lambda_1^k > 0 \;\Rightarrow\; x_1^* = 0 \quad \text{and} \quad \lambda_1^k < 0 \;\Rightarrow\; x_1^* = 1$$

in any optimal solution x^* of **P**!

... which provides a rationale for the following

"joint+marginal" algorithm for 0/1 optimization

While $i < n$ repeat:

- Consider the 0/1 problem $\mathbf{P}(x_1, \ldots, x_{i-1})$ which is **P** where the first $i-1$ components of $\mathbf{x}$ are already fixed.
- Solve the k-th semidefinite relaxation with parameter x_i associated with $\mathbf{P}(x_1, \ldots, x_{i-1})$, and get an optimal solution $(\lambda_0^k, \lambda_1^k, \sigma_j^k)$ of the dual.
- If $\lambda_1^k > 0$ set $x_i = 0$ and $x_i = 1$ otherwise.
- If $\mathbf{P}(x_1, \ldots, x_{i-1}, x_i)$ has a feasible solution select $x_i := x_i$ and $x_i := 1 - x_i$ otherwise.

Hence .. if k is suffciently large,

$$\lambda_1^k > 0 \Rightarrow x_1^* = 0 \quad \text{and} \quad \lambda_1^k < 0 \Rightarrow x_1^* = 1$$

in any optimal solution x^* of **P**!

... which provides a rationale for the following

"joint+marginal" algorithm for 0/1 optimization

While $i < n$ repeat:

- Consider the 0/1 problem $\mathbf{P}(x_1, \ldots, x_{i-1})$ which is **P** where the first $i - 1$ components of $\mathbf{x}$ are already fixed.
- Solve the k-th semidefinite relaxation with parameter x_i associated with $\mathbf{P}(x_1, \ldots, x_{i-1})$, and get an optimal solution $(\lambda_0^k, \lambda_1^k, \sigma_j^k)$ of the dual.
- If $\lambda_1^k > 0$ set $x_i = 0$ and $x_i = 1$ otherwise.
- If $\mathbf{P}(x_1, \ldots, x_{i-1}, x_i)$ has a feasible solution select $x_i := x_i$ and $x_i := 1 - x_i$ otherwise.

Hence .. if k is suffciently large,

$$\lambda_1^k > 0 \Rightarrow x_1^* = 0 \quad \text{and} \quad \lambda_1^k < 0 \Rightarrow x_1^* = 1$$

in any optimal solution x^* of **P**!

... which provides a rationale for the following

"joint+marginal" algorithm for 0/1 optimization

While $i < n$ repeat:

- Consider the 0/1 problem $\mathbf{P}(x_1, \ldots, x_{i-1})$ which is **P** where the first $i-1$ components of $\mathbf{x}$ are already fixed.
- Solve the k-th semidefinite relaxation with parameter x_i associated with $\mathbf{P}(x_1, \ldots, x_{i-1})$, and get an optimal solution $(\lambda_0^k, \lambda_1^k, \sigma_j^k)$ of the dual.
- If $\lambda_1^k > 0$ set $x_i = 0$ and $x_i = 1$ otherwise.
- If $\mathbf{P}(x_1, \ldots, x_{i-1}, x_i)$ has a feasible solution select $x_i := x_i$ and $x_i := 1 - x_i$ otherwise.

Hence .. if k is suffciently large,

$$\lambda_1^k > 0 \Rightarrow x_1^* = 0 \quad \text{and} \quad \lambda_1^k < 0 \Rightarrow x_1^* = 1$$

in any optimal solution x^* of **P**!

... which provides a rationale for the following

"joint+marginal" algorithm for 0/1 optimization

While $i < n$ repeat:

- Consider the 0/1 problem $\mathbf{P}(x_1, \ldots, x_{i-1})$ which is **P** where the first $i - 1$ components of $\mathbf{x}$ are already fixed.
- Solve the k-th semidefinite relaxation with parameter x_i associated with $\mathbf{P}(x_1, \ldots, x_{i-1})$, and get an optimal solution $(\lambda_0^k, \lambda_1^k, \sigma_j^k)$ of the dual.
- If $\lambda_1^k > 0$ set $x_i = 0$ and $x_i = 1$ otherwise.
- If $\mathbf{P}(x_1, \ldots, x_{i-1}, x_i)$ has a feasible solution select $x_i := x_i$ and $x_i := 1 - x_i$ otherwise.

The rationale behind the "joint+marginal" algorithm:

- The larger k, the better the approximation of $J(\mathbf{y})$ by the univariate polynomial $J_k(\mathbf{y}) = \lambda_0^k + \lambda_1^k y$. And so in minimizing $J_k(\mathbf{y})$ over $\mathbf{Y}$ one has a good chance to obtain $x_1 \approx x_1^*$, where $\mathbf{x}^*$ is a global minimizer of $\mathbf{P}$. And so at the end one may expect $\mathbf{x} \approx \mathbf{x}^*$.
- But ... the interest is to precisely have k not too large so as to handle relatively large size problems.
- Preliminary results are encouraging!

The rationale behind the "joint+marginal" algorithm:

- The larger k, the better the approximation of $J(\mathbf{y})$ by the univariate polynomial $J_k(\mathbf{y}) = \lambda_0^k + \lambda_1^k y$. And so in minimizing $J_k(\mathbf{y})$ over $\mathbf{Y}$ one has a good chance to obtain $x_1 \approx x_1^*$, where $\mathbf{x}^*$ is a global minimizer of **P**. And so at the end one may expect $\mathbf{x} \approx \mathbf{x}^*$.
- But ... the interest is to precisely have k not too large so as to handle relatively large size problems.
- Preliminary results are encouraging!

EX: MAX-CUT problem: $\max \mathbf{x}^T \mathbf{A} \mathbf{x} \,:\, \mathbf{x} \in \{-1,1\}^n\}$

• $\mathbf{Y} = \{-1, 1\}$, and let $\varphi(\{1\}) = \varphi(\{-1\} = 1/2$.

• We fix $k = 1$. The semidefinite program

$$\mathbf{Q}_1 : \quad \left\{ \begin{array}{ll} \max & \mathrm{trace}(A\,X) \\ \text{s.t.} & \left(\begin{array}{cc} 1 & x' \\ x & X \end{array} \right) \succeq 0 \\ & X_{ii} = 1, \quad i = 1, \ldots, n \end{array} \right.$$

is the first semidefinite relaxation of **P** (with celebrated Goemans & Williamson performance garantee).

• The 1-th parametric semidefinite relaxation reads:

$$\left\{ \begin{array}{ll} \rho_1 = \max & \mathrm{trace}(A\,X) \\ \text{s.t.} & \left(\begin{array}{cc} 1 & x' \\ x & X \end{array} \right) \succeq 0 \\ & X_{ii} = 1, \quad i = 1, \ldots, n \\ & x_1 = 1/2 \end{array} \right.$$

which is $\mathbf{Q}_1$ with ONE additional constraint $x_1 = 1/2$.

EX: MAX-CUT problem: $\max \mathbf{x}^T \mathbf{A} \mathbf{x} \; : \; \mathbf{x} \in \{-1, 1\}^n\}$

- $\mathbf{Y} = \{-1, 1\}$, and let $\varphi(\{1\}) = \varphi(\{-1\} = 1/2$.
- We fix $k = 1$. The semidefinite program

$$\mathbf{Q}_1 : \quad \left\{ \begin{array}{ll} \max & \mathrm{trace}(A\, X) \\ \text{s.t.} & \left(\begin{array}{cc} 1 & x' \\ x & X \end{array} \right) \succeq 0 \\ & X_{ii} = 1, \quad i = 1, \ldots, n \end{array} \right.$$

is the first semidefinite relaxation of **P** (with celebrated Goemans & Williamson performance garantee).

- The 1-th parametric semidefinite relaxation reads:

$$\left\{ \begin{array}{rl} \rho_1 = \max & \mathrm{trace}(A\, X) \\ \text{s.t.} & \left(\begin{array}{cc} 1 & x' \\ x & X \end{array} \right) \succeq 0 \\ & X_{ii} = 1, \quad i = 1, \ldots, n \\ & x_1 = 1/2 \end{array} \right.$$

which is $\mathbf{Q}_1$ with ONE additional constraint $x_1 = 1/2$.

We have tested the "joint+marginal" algorithm on a sample of 50 and 100 randomly generated MAXCUT instances with $n = 20, 30$ and 40 nodes in the corresponding graph. An arc (i, j) is generated with probability $1/2$. and A_{ij} is generated according to a uniform distribution on $[0, 10]$.
Let $\mathbf{P}_1$ be the values of the solution generated by the "joint+marginal" algorithm.

n	20	30	40
$(\mathbf{P}_1 - \mathbf{Q}_1)/\lvert\mathbf{Q}_1\rvert$	10.3%	12.3%	12.5%

Table: Relative error for MAXCUT

† We implemented the "max-gap" variant which instead of selecting x_1, then x_2, etc. selects first the variable x_i with maximum gap $|J_1(-1) - J_1(1)|$, etc.

We have tested the "joint+marginal" algorithm on a sample of 50 and 100 randomly generated MAXCUT instances with $n = 20, 30$ and 40 nodes in the corresponding graph. An arc (i, j) is generated with probability $1/2$. and A_{ij} is generated according to a uniform distribution on $[0, 10]$.
Let $\mathbf{P}_1$ be the values of the solution generated by the "joint+marginal" algorithm.

n	20	30	40
$(\mathbf{P}_1 - \mathbf{Q}_1)/\lvert\mathbf{Q}_1\rvert$	10.3%	12.3%	12.5%

Table: Relative error for MAXCUT

† We implemented the "max-gap" variant which instead of selecting x_1, then x_2, etc. selects first the variable x_i with maximum gap $|J_1(-1) - J_1(1)|$, etc.

$$\mathbf{P}: \quad \max_{\mathbf{x}} \left\{ \sum_{i=1}^{n} c_i\, x_i \;:\; \sum_{i=1}^{n} a_i\, x_i \,\leq\, b\,; \quad \mathbf{x} \in \{0,1\}^n \right\}$$

The first semidefinite relaxation of **P** is:

$$\mathbf{Q}_1 : \quad \left\{ \begin{array}{rl} \max & \mathrm{trace}(A\,X) \\ \text{s.t.} & \begin{pmatrix} 1 & x' \\ x & X \end{pmatrix} \succeq 0 \\ & X_{ii} = 1, \quad i = 1,\ldots,n \\ & \sum_{i=1}^{n} a_i\, x_i \leq b \end{array} \right.$$

For the 1-th parametric semidefinite relaxation it suffices to add the linear constraint $x_1 = 1/2$.

0/1 knapsack problem

$$\mathbf{P}: \quad \max_{\mathbf{x}} \left\{ \sum_{i=1}^{n} c_i\, x_i \;:\; \sum_{i=1}^{n} a_i\, x_i \,\leq\, b\,; \quad \mathbf{x} \in \{0,1\}^n \right\}$$

The first semidefinite relaxation of **P** is:

$$\mathbf{Q}_1: \quad \left\{ \begin{array}{ll} \max & \mathrm{trace}(A\,X) \\ \text{s.t.} & \left(\begin{array}{cc} 1 & x' \\ x & X \end{array} \right) \succeq 0 \\ & X_{ii} = 1, \quad i = 1, \ldots, n \\ & \sum_{i=1}^{n} a_i\, x_i \leq b \end{array} \right.$$

For the 1-th parametric semidefinite relaxation it suffices to add the linear constraint $x_1 = 1/2$.

One may even add the n redundant constraints:

$$x_\ell \left[\sum_{i=1}^n a_i\, x_i - b\right] \leq 0 \qquad \ell = 1, \ldots, n,$$

which read

$$\sum_{i=1}^n a_i\, X_{\ell i} - b\, x_\ell \leq 0 \qquad \ell = 1, \ldots, n,$$

We have tested the "joint+marginal" algorithm on a sample of problems with $n = 40$ and $n = 50$ variables where:

• $b = \sum_i a_i/2$, and the integers a_i's are generated uniformly in $[10, 100]$.

• The vector $\mathbf{c}$ is generated by: $c_i = s * \epsilon + a_i$ with $s = 0.1, 1, 5, 10$ and ϵ is a random variable uniformly distributed in $[0, 1]$.

s	0.1	1	5	10
$(\mathbf{P}_1 - \mathbf{Q}_1)/\lvert\mathbf{Q}_1\rvert$	2.5%	1.42%	1.5%	

Table: Relative error for 0/1 KNAPSACK: $n = 40$

s	0.1	1	5	10
$(\mathbf{P}_1 - \mathbf{Q}_1)/\lvert\mathbf{Q}_1\rvert$	1.86%	1.42%	0.7%	0.08%

Table: Relative error for 0/1 KNAPSACK: $n = 50$

s	0.1	1	5	10
$(P_1 - Q_1)/\lvert Q_1 \rvert$	2.5%	1.42%	1.5%	

Table: Relative error for 0/1 KNAPSACK: $n = 40$

s	0.1	1	5	10
$(P_1 - Q_1)/\lvert Q_1 \rvert$	1.86%	1.42%	0.7%	0.08%

Table: Relative error for 0/1 KNAPSACK: $n = 50$

k-cluster problem

$$\mathbf{P}: \quad \max_{\mathbf{x}} \left\{ \sum_{i=1}^{n} \mathbf{x}'\mathbf{A}\mathbf{x} \; : \; \sum_{i=1}^{n} x_i = k; \quad \mathbf{x} \in \{0,1\}^n \right\}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$ is real symmetric matrix associated with a graph. The first semidefinite relaxation of **P** is:

$$\mathbf{Q}_1: \quad \left\{ \begin{array}{rl} \max & \mathrm{trace}(A\,X) \\ \text{s.t.} & \begin{pmatrix} 1 & x' \\ x & X \end{pmatrix} \succeq 0 \\ & X_{ii} = 1, \quad i = 1,\ldots,n \\ & \sum_{i=1}^{n} x_i = k \end{array} \right.$$

For the 1-parametric semidefinite relaxation it suffices to add the linear constraint $x_1 = 1/2$.

$$\mathbf{P}: \quad \max_{\mathbf{x}} \left\{ \sum_{i=1}^{n} \mathbf{x}'\mathbf{A}\mathbf{x} \; : \; \sum_{i=1}^{n} x_i \,=\, k\,; \quad \mathbf{x} \in \{0,1\}^n \right\}$$

where $\mathbf{A} \in \mathbb{R}^{n\times n}$ is real symmetric matrix associated with a graph. The first semidefinite relaxation of **P** is:

$$\mathbf{Q}_1 : \quad \left\{ \begin{array}{ll} \max & \mathrm{trace}(A\,X) \\ \text{s.t.} & \begin{pmatrix} 1 & x' \\ x & X \end{pmatrix} \succeq 0 \\ & X_{ii} = 1, \quad i = 1,\ldots,n \\ & \sum_{i=1}^{n} x_i = k \end{array} \right.$$

For the 1-parametric semidefinite relaxation it suffices to add the linear constraint $x_1 = 1/2$.

One may even add the n redundant constraints:

$$x_\ell \left[\sum_{i=1}^n x_i - k\right] = 0 \qquad \ell = 1, \ldots, n,$$

which read

$$\sum_{i=1}^n X_{\ell i} - k\, x_\ell = 0 \qquad \ell = 1, \ldots, n,$$

We have tested the "joint+marginal" algorithm on a sample of problems with $n = 40$ nodes, with $k = n/2$ and where an arc (i, j) is generated with probability $1/2$. and A_{ij} is generated according to a uniform distribution on $[0, 10]$.

On 4 problems we observed an average relative error $(\mathbf{P}_1 - \mathbf{Q}_1)/|\mathbf{Q}_1|$ of less than 5%.

We have tested the "joint+marginal" algorithm on a sample of problems with $n = 40$ nodes, with $k = n/2$ and where an arc (i, j) is generated with probability $1/2$. and A_{ij} is generated according to a uniform distribution on $[0, 10]$.

On 4 problems we observed an average relative error $(\mathbf{P}_1 - \mathbf{Q}_1)/|\mathbf{Q}_1|$ of less than 5%.

THANK YOU !!