Nonlinear Mixed-Integer Optimization

MINO/COST SPRING SCHOOL ON OPTIMIZATION

Jon Lee — Andrea Lodi

University of Michigan — University of Bologna

10 April 2014

Outline for the Day

- 9:15-10:15. lecture (Lee): Introduction to MINLP // Complexity of MINLP: Hardness and polynomial tractability
- 10:15-11:00. coffee break
- 11:00-12:00. lecture (Lodi): General-purpose algorithms for convex and non-convex MINLP
- 12:00-14:15. lunch
- 14:15-15:00. lecture (Lee): Non-convex quadratic MINLP
- 15:00-15:30. coffee break
- 15:30-16:15. lecture (Lodi): Software and computational advances
- 16:15-16:30. short break
- 16:30-17:00. problem session

O - Introduction to MINLP

I - Complexity of MINLP

Hardness and polynomial tractability

Nonlinear Integer/Discrete Optimization

$$\max / \min \{ f(x) \ : \ x \in \mathcal{F} \},$$

where $f : \mathbb{R}^n \Rightarrow \mathbb{R}$, and $\mathcal{F}$ is countable (and often finite).

Nonlinear Integer/Discrete Optimization

$$\max / \min \{ \ f(x) \ : \ x \in \mathcal{F} \ \},$$

where $f : \mathbb{R}^n \Rightarrow \mathbb{R}$, and $\mathcal{F}$ is countable (and often finite).

Immediate questions:

- What are the applications?

Nonlinear Integer/Discrete Optimization

$$\max / \min \{ f(x) \ : \ x \in \mathcal{F} \},$$

where $f : \mathbb{R}^n \Rightarrow \mathbb{R}$, and $\mathcal{F}$ is countable (and often finite).

Immediate questions:

- What are the applications? What are NOT the applications?!
  - $\max / \min_{(x,y)} \{ g(x,y) \ : \ h(x,y) \leq 0, \ x \in \mathcal{F} \} \Rightarrow$
    For $x \in \mathcal{F}$, let $f(x) := \max / \min_y \{ g(x,y) \ : \ h(x,y) \leq 0 \}$

Nonlinear Integer/Discrete Optimization

$$\max / \min \{ \ f(x) \ : \ x \in \mathcal{F} \ \},$$

where $f : \mathbb{R}^n \Rightarrow \mathbb{R}$, and $\mathcal{F}$ is countable (and often finite).

Immediate questions:

- What are the applications? What are NOT the applications?!
  - $\max / \min_{(x,y)} \{ \ g(x,y) \ : \ h(x,y) \leq 0, \ x \in \mathcal{F} \ \} \Rightarrow$
    For $x \in \mathcal{F}$, let $f(x) := \max / \min_y \{ \ g(x,y) \ : \ h(x,y) \leq 0 \ \}$
- What kind of objectives f are we interested in? How do we receive f ?

Nonlinear Integer/Discrete Optimization

$$\max / \min \{ f(x) \ : \ x \in \mathcal{F} \},$$

where $f : \mathbb{R}^n \Rightarrow \mathbb{R}$, and $\mathcal{F}$ is countable (and often finite).

Immediate questions:

- What are the applications? What are NOT the applications?!
  - $\max / \min_{(x,y)} \{ g(x,y) \ : \ h(x,y) \leq 0, \ x \in \mathcal{F} \} \Rightarrow$
    For $x \in \mathcal{F}$, let $f(x) := \max / \min_y \{ g(x,y) \ : \ h(x,y) \leq 0 \}$
- What kind of objectives f are we interested in? How do we receive f ?
- What kind of feasibility sets $\mathcal{F}$ are we interested in? How do we receive $\mathcal{F}$?

Nonlinear Integer/Discrete Optimization

$$\max / \min \{ f(x) \ : \ x \in \mathcal{F} \},$$

where $f : \mathbb{R}^n \Rightarrow \mathbb{R}$, and $\mathcal{F}$ is countable (and often finite).

Immediate questions:

- What are the applications? What are NOT the applications?!
  - $\max / \min_{(x,y)} \{ g(x,y) \ : \ h(x,y) \leq 0, \ x \in \mathcal{F} \} \Rightarrow$
    For $x \in \mathcal{F}$, let $f(x) := \max / \min_y \{ g(x,y) \ : \ h(x,y) \leq 0 \}$
- What kind of objectives f are we interested in? How do we receive f ?
- What kind of feasibility sets $\mathcal{F}$ are we interested in? How do we receive $\mathcal{F}$?
- What are we trying to achieve? Theory? Practicality?

What can we hope to achieve?

- An optimal solution (and an optimality certificate)

What can we hope to achieve?

- An optimal solution (and an optimality certificate)
- The O.R./Math-Opt offering: A feasible solution and an optimality bound — no *a priori* bound on the bound, and often no efficiency guarantee

What can we hope to achieve?

- An optimal solution (and an optimality certificate)
- The O.R./Math-Opt offering: A feasible solution and an optimality bound — no *a priori* bound on the bound, and often no efficiency guarantee
- The Theoretical C.S. offering: A feasible solution with an *a priori* approximation guarantee (e.g., a $(1-\epsilon)$-approximation or a k-best solution) and an efficiency guarantee

What can we hope to achieve?

- An optimal solution (and an optimality certificate)
- The O.R./Math-Opt offering: A feasible solution and an optimality bound — no *a priori* bound on the bound, and often no efficiency guarantee
- The Theoretical C.S. offering: A feasible solution with an *a priori* approximation guarantee (e.g., a $(1 - \epsilon)$-approximation or a k-best solution) and an efficiency guarantee
- A publication

What can we hope to achieve?

- An optimal solution (and an optimality certificate)
- The O.R./Math-Opt offering: A feasible solution and an optimality bound — no *a priori* bound on the bound, and often no efficiency guarantee
- The Theoretical C.S. offering: A feasible solution with an *a priori* approximation guarantee (e.g., a $(1-\epsilon)$-approximation or a k-best solution) and an efficiency guarantee
- A publication

What solution time can we afford?

- Theory: Polynomial time? — with respect to to what kind of encoding?

What can we hope to achieve?

- An optimal solution (and an optimality certificate)
- The O.R./Math-Opt offering: A feasible solution and an optimality bound — no *a priori* bound on the bound, and often no efficiency guarantee
- The Theoretical C.S. offering: A feasible solution with an *a priori* approximation guarantee (e.g., a $(1-\epsilon)$-approximation or a k-best solution) and an efficiency guarantee
- A publication

What solution time can we afford?

- Theory: Polynomial time? — with respect to to what kind of encoding?
- Practice: A few seconds? minutes? hours? days? weeks?

Applications to drive advances in methodology

Applications to drive advances in methodology

- Unit commitment problems with ramp-up costs; separable convex continuous nonlinearities — a first step up from MILP.

Applications to drive advances in methodology

- Unit commitment problems with ramp-up costs; separable convex continuous nonlinearities — a first step up from MILP.
- (Water and gas) delivery network planning with physics; difficult continuous nonlinearities.

Applications to drive advances in methodology

- Unit commitment problems with ramp-up costs; separable convex continuous nonlinearities — a first step up from MILP.
- (Water and gas) delivery network planning with physics; difficult continuous nonlinearities.
- Portfolio optimization with cardinality constraints; (convex?) continuous quadratics and sparse optimization.

Applications to drive advances in methodology

- Unit commitment problems with ramp-up costs; separable convex continuous nonlinearities — a first step up from MILP.
- (Water and gas) delivery network planning with physics; difficult continuous nonlinearities.
- Portfolio optimization with cardinality constraints; (convex?) continuous quadratics and sparse optimization.
- Combinatorial optimization on graphs; quadratic binary problems.

Applications to drive advances in methodology

- Unit commitment problems with ramp-up costs; separable convex continuous nonlinearities — a first step up from MILP.
- (Water and gas) delivery network planning with physics; difficult continuous nonlinearities.
- Portfolio optimization with cardinality constraints; (convex?) continuous quadratics and sparse optimization.
- Combinatorial optimization on graphs; quadratic binary problems.
- Microelectronics imaging some distance-geometry problems; global-optimization of low-order polynomials.

Applications to drive advances in methodology

- Unit commitment problems with ramp-up costs; separable convex continuous nonlinearities — a first step up from MILP.
- (Water and gas) delivery network planning with physics; difficult continuous nonlinearities.
- Portfolio optimization with cardinality constraints; (convex?) continuous quadratics and sparse optimization.
- Combinatorial optimization on graphs; quadratic binary problems.
- Microelectronics imaging some distance-geometry problems; global-optimization of low-order polynomials.
- Difficult counting functions; rank of a matrix.

Applications to drive advances in methodology

- Unit commitment problems with ramp-up costs; separable convex continuous nonlinearities — a first step up from MILP.
- (Water and gas) delivery network planning with physics; difficult continuous nonlinearities.
- Portfolio optimization with cardinality constraints; (convex?) continuous quadratics and sparse optimization.
- Combinatorial optimization on graphs; quadratic binary problems.
- Microelectronics imaging some distance-geometry problems; global-optimization of low-order polynomials.
- Difficult counting functions; rank of a matrix.
- Submodular optimization (a bridge from cuts to difficult inseparable and black-box functions).

Applications to drive advances in methodology

- Unit commitment problems with ramp-up costs; separable convex continuous nonlinearities — a first step up from MILP.
- (Water and gas) delivery network planning with physics; difficult continuous nonlinearities.
- Portfolio optimization with cardinality constraints; (convex?) continuous quadratics and sparse optimization.
- Combinatorial optimization on graphs; quadratic binary problems.
- Microelectronics imaging some distance-geometry problems; global-optimization of low-order polynomials.
- Difficult counting functions; rank of a matrix.
- Submodular optimization (a bridge from cuts to difficult inseparable and black-box functions).

... Advances in methodology to make applied impact

A prologue of approaches

- Heuristic approach (driven mostly by “local” ideas to quickly find good solutions to practical instances)

A prologue of approaches

- Heuristic approach (driven mostly by "local" ideas to quickly find good solutions to <u>practical</u> instances)
- Algorithmic engineering driven by mathematics, with the solution of <u>practical</u> instances in mind:

A prologue of approaches

- Heuristic approach (driven mostly by “local” ideas to quickly find good solutions to practical instances)
- Algorithmic engineering driven by mathematics, with the solution of practical instances in mind:
  - Refining and subdividing relaxations (employ convex geometry of inequalities in real variables)

A prologue of approaches

- Heuristic approach (driven mostly by "local" ideas to quickly find good solutions to practical instances)
- Algorithmic engineering driven by mathematics, with the solution of practical instances in mind:
  - Refining and subdividing relaxations (employ convex geometry of inequalities in real variables)
    - Operations-research approach I: B&B, Outer approx, hybrid approaches [aimed at finding optima of convex instances] (e.g., `Bonmin`)

A prologue of approaches

- Heuristic approach (driven mostly by "local" ideas to quickly find good solutions to practical instances)
- Algorithmic engineering driven by mathematics, with the solution of practical instances in mind:
  - Refining and subdividing relaxations (employ convex geometry of inequalities in real variables)
    - ⋆ Operations-research approach I: B&B, Outer approx, hybrid approaches [aimed at finding optima of convex instances] (e.g., `Bonmin`)
    - ⋆ Operations-research approach II: Spatial B&B/Global optimization [aimed at finding optima of nonconvex instances] (e.g., `Baron`, `Couenne`)

A prologue of approaches

- Heuristic approach (driven mostly by "local" ideas to quickly find good solutions to practical instances)
- Algorithmic engineering driven by mathematics, with the solution of practical instances in mind:
  - Refining and subdividing relaxations (employ convex geometry of inequalities in real variables)
    - Operations-research approach I: B&B, Outer approx, hybrid approaches [aimed at finding optima of convex instances] (e.g., `Bonmin`)
    - Operations-research approach II: Spatial B&B/Global optimization [aimed at finding optima of nonconvex instances] (e.g., `Baron`, `Couenne`)
    - May hybridize these OR approaches with CP techniques

A prologue of approaches

- Heuristic approach (driven mostly by "local" ideas to quickly find good solutions to practical instances)
- Algorithmic engineering driven by mathematics, with the solution of practical instances in mind:
  - Refining and subdividing relaxations (employ convex geometry of inequalities in real variables)
    - Operations-research approach I: B&B, Outer approx, hybrid approaches [aimed at finding optima of convex instances] (e.g., `Bonmin`)
    - Operations-research approach II: Spatial B&B/Global optimization [aimed at finding optima of nonconvex instances] (e.g., `Baron`, `Couenne`)
    - May hybridize these OR approaches with CP techniques
  - Algebraic approach: Model via a system of polynomial equations in complex variables which is attacked via methods of algebraic geometry (e.g., De Loera, Lee, Margulies, Malkin, Onn have had success on graph coloring problems with up to 2000 vertices)

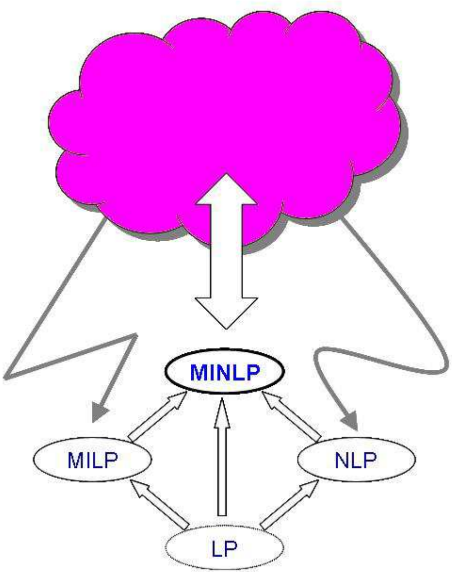
A prologue of approaches

- Heuristic approach (driven mostly by “local” ideas to quickly find good solutions to practical instances)
- Algorithmic engineering driven by mathematics, with the solution of practical instances in mind:
  - Refining and subdividing relaxations (employ convex geometry of inequalities in real variables)
    - Operations-research approach I: B&B, Outer approx, hybrid approaches [aimed at finding optima of convex instances] (e.g., `Bonmin`)
    - Operations-research approach II: Spatial B&B/Global optimization [aimed at finding optima of nonconvex instances] (e.g., `Baron`, `Couenne`)
    - May hybridize these OR approaches with CP techniques
  - Algebraic approach: Model via a system of polynomial equations in complex variables which is attacked via methods of algebraic geometry (e.g., De Loera, Lee, Margulies, Malkin, Onn have had success on graph coloring problems with up to 2000 vertices)
- Theoretical computer science approach: restrict problem classes to find efficient exact and approximation algorithms or prove hardness...

A prologue of approaches

- Heuristic approach (driven mostly by "local" ideas to quickly find good solutions to practical instances)
- Algorithmic engineering driven by mathematics, with the solution of practical instances in mind:
  - Refining and subdividing relaxations (employ convex geometry of inequalities in real variables)
    - Operations-research approach I: B&B, Outer approx, hybrid approaches [aimed at finding optima of convex instances] (e.g., `Bonmin`)
    - Operations-research approach II: Spatial B&B/Global optimization [aimed at finding optima of nonconvex instances] (e.g., `Baron`, `Couenne`)
    - May hybridize these OR approaches with CP techniques
  - Algebraic approach: Model via a system of polynomial equations in complex variables which is attacked via methods of algebraic geometry (e.g., De Loera, Lee, Margulies, Malkin, Onn have had success on graph coloring problems with up to 2000 vertices)
- Theoretical computer science approach: restrict problem classes to find efficient exact and approximation algorithms or prove hardness...keeping in mind the applications!

An aside: The mathematical optimization (O.R.) view



Classification of nonlinear discrete optimization

- Objective function:

LINEAR $\Longrightarrow$ MILDLY NONLINEAR $\Longrightarrow$ VERY NONLINEAR (EXPLICIT) $\Longrightarrow$ "ARBITRARY" NONLINEAR (ORACLE? SIMULATION?)

Classification of nonlinear discrete optimization

- Objective function:

  LINEAR $\Longrightarrow$ MILDLY NONLINEAR $\Longrightarrow$ VERY NONLINEAR (EXPLICIT) $\Longrightarrow$ "ARBITRARY" NONLINEAR (ORACLE? SIMULATION?)

- Encoding:

  UNARY ENCODING; $a_1, \ldots, a_p$-ENCODING $\Longrightarrow$ BINARY ENCODING

Classification of nonlinear discrete optimization

- Objective function:

  LINEAR $\Longrightarrow$ MILDLY NONLINEAR $\Longrightarrow$ VERY NONLINEAR (EXPLICIT) $\Longrightarrow$ "ARBITRARY" NONLINEAR (ORACLE? SIMULATION?)

- Encoding:

  UNARY ENCODING; $a_1, \ldots, a_p$-ENCODING $\Longrightarrow$ BINARY ENCODING

- Feasible set description:

  STRUCTURED CONSTRAINTS (EXPLICIT? ORACLE?) $\Longrightarrow$ UNSTRUCTURED CONSTRAINTS (EXPLICIT? ORACLE?/ SIMULATION?)

Reduced-dimension nonlinear discrete opt

Given finite $\mathcal{F} \subset \mathbb{Z}^n$, weight matrix $W \in \mathbb{Z}^{d \times n}$ and function $f : \mathbb{R}^d \to \mathbb{R}$, solve

$$P(\mathcal{F}, f, W) : \qquad \min / \max \{f(Wx) \ : \ x \in \mathcal{F}\}$$

Motivation is multi-objective optimization, where f trades off the linear functions describes by the rows of W

Reduced-dimension nonlinear discrete opt

Given finite $\mathcal{F} \subset \mathbb{Z}^n$, weight matrix $W \in \mathbb{Z}^{d \times n}$ and function $f : \mathbb{R}^d \to \mathbb{R}$, solve

$$P(\mathcal{F}, f, W) : \qquad \min / \max \{ f(Wx) \ : \ x \in \mathcal{F} \}$$

Motivation is multi-objective optimization, where f trades off the linear functions describes by the rows of W

Typical assumptions to gain theoretical efficiency:

- fixed d ($d \leq n$)
- f given by a ‘comparison oracle’
- encoding of W:
    - unary encoded
    - $W_{i,j} \in \{a_1, \ldots, a_p\}$ (p fixed, a_i binary-encoded positive integers)
    - generalized unary: $\sum_{i=1}^{p} \lambda_i a_i$, with λ_i unary encoded
- $\mathcal{F}$ given via different oracles:
    - (poly)matroids,
    - multi-knapsacks
    - matchings
- $\mathcal{F} \subset \{ x \in Z^n \ : \ -\beta \mathbf{1} \leq x \leq \beta \mathbf{1} \}$, unary encoded β

Hardness and incomputability for polynomial opt

Hardness and incomputability for polynomial opt

Theorem (Max Cut)

Pure continuous polynomial optimization over polytopes in varying dimension is NP-hard. Moreover (Hastad), there does not exist a fully polynomial-time approximation scheme (FPTAS) (unless $\mathrm{P} = \mathrm{NP}$*)*

Hardness and incomputability for polynomial opt

Theorem (Max Cut)

Pure continuous polynomial optimization over polytopes in varying dimension is NP-hard. Moreover (Hastad), there does not exist a fully polynomial-time approximation scheme (FPTAS) (unless $\mathrm{P} = \mathrm{NP}$*)*

Theorem (see De Loera, Hemmecke, Köppe, Weismantel)

The problem of minimizing a degree-4 polynomial over the lattice points of a convex polygon is NP-hard.

Hardness and incomputability for polynomial opt

Theorem (Max Cut)

Pure continuous polynomial optimization over polytopes in varying dimension is NP-hard. Moreover (Hastad), there does not exist a fully polynomial-time approximation scheme (FPTAS) (unless $\mathrm{P} = \mathrm{NP}$*)*

Theorem (see De Loera, Hemmecke, Köppe, Weismantel)

The problem of minimizing a degree-4 polynomial over the lattice points of a convex polygon is NP-hard.

Theorem (Jeroslow)

The problem of minimizing a linear form in integer variables over quadratic constraints is not computable by a recursive function.

Hardness and incomputability for polynomial opt

Theorem (Max Cut)

Pure continuous polynomial optimization over polytopes in varying dimension is NP-hard. Moreover (Hastad), there does not exist a fully polynomial-time approximation scheme (FPTAS) (unless $\mathrm{P} = \mathrm{NP}$*)*

Theorem (see De Loera, Hemmecke, Köppe, Weismantel)

The problem of minimizing a degree-4 polynomial over the lattice points of a convex polygon is NP-hard.

Theorem (Jeroslow)

The problem of minimizing a linear form in integer variables over quadratic constraints is not computable by a recursive function.

Theorem (see De Loera, Hemmecke, Köppe, Weismantel)

The problem of minimizing a linear form in at most 10 integer variables over polynomial constraints is not computable by a recursive function.

Intractability of reduced-dimension nonlinear discrete opt

- $d = 1$, binary-encoded W is provably intractable (even for $W := (1, 2, 4, \ldots, 2^{n-1})$, f given by a comparison oracle, $\mathcal{F}$ the incidence vectors of bases of a graphic or uniform matroid)
- $d = n$, $W = I$ is provably intractable (even for f given by a comparison oracle, $\mathcal{F}$ the incidence vectors of bases of a graphic or uniform matroid)

In both cases we can have Wx and hence $f(Wx)$ take on a distinct value for each $x \in \mathcal{F}$, and $|\mathcal{F}|$ is exponentially large

Intractability of reduced-dimension nonlinear discrete opt

- $d = 1$, binary-encoded W is provably intractable (even for $W := (1, 2, 4, \ldots, 2^{n-1})$, f given by a comparison oracle, $\mathcal{F}$ the incidence vectors of bases of a graphic or uniform matroid)
- $d = n$, $W = I$ is provably intractable (even for f given by a comparison oracle, $\mathcal{F}$ the incidence vectors of bases of a graphic or uniform matroid)

IN BOTH CASES WE CAN HAVE Wx AND HENCE $f(Wx)$ TAKE ON A DISTINCT VALUE FOR EACH $x \in \mathcal{F}$, AND $|\mathcal{F}|$ IS EXPONENTIALLY LARGE

Given binary encoded $u \in \mathbb{Z}$ and $W \in \mathbb{Z}^{1 \times n}$, it is *NP*-complete to determine

- if there is an $x \in \mathcal{F}$ such that $Wx = u$, or
- whether the minimum, over $x \in \mathcal{F}$, of the simple convex function $f(x) = (Wx - u)^2$ is zero,

(even for $\mathcal{F}$ the incidence vectors of bases of a graphic or uniform matroid)

Well-described independence systems

Definition

$\mathcal{F}$ is **well described** (via linear inequalities) in the sense of GLS
$\equiv$ **linear optimization** over $\mathcal{F}$ can be done efficiently

Definition

$\mathcal{F} \subseteq \{0,1\}^n$ is an **independence system** if for $x, y \in \{0,1\}^n$,

$$x \leq y \in \mathcal{F} \qquad \Longrightarrow \qquad x \in \mathcal{F} \ .$$

Example

- forests of a graph, independent sets of a matroid
- (poly)matroids
- matchings of a graph
- small multi-knapsacks
- stable sets of certain graphs (e.g., perfect $\supset$ bipartite; claw-free $\supset$ quasi-line $\supset$ line)

Intractability

Theorem (Lee, Onn, Weismantel '09)

There is <u>*no efficient algorithm*</u> *for computing an optimal solution of the* <u>*one-dimensional*</u> *nonlinear optimization problem* $\min\{f(w'x) : x \in \mathcal{F}\}$ *over a well-described independence system,* $\mathcal{F}$*, with* f *presented by a comparison oracle, and single weight vector* $w \in \{2,3\}^n$ *.*

Intractability

Theorem (Lee, Onn, Weismantel '09)

There is no efficient algorithm for computing an optimal solution of the one-dimensional nonlinear optimization problem $\min\{f(w'x) : x \in \mathcal{F}\}$ *over a well-described independence system,* $\mathcal{F}$*, with* f *presented by a comparison oracle, and single weight vector* $w \in \{2,3\}^n$ *.*

Theorem (Lee, Onn, Weismantel '11)

There is a universal constant ρ *such that no efficient algorithm can compute a "*ρn*-best solution" of the 2-dimensional nonlinear discrete optimization problem* $\min\{f(Wx) : x \in S\}$ *over every well-described independence system* $\mathcal{F} \subseteq \{0,1\}^n$*, with* W *an integer* $2 \times n$ *weight matrix each column of which is one of the unit vectors in* $\mathbb{Z}^2$*, and* f *an explicit function supported on* $\{0,1,\ldots,n\}^2$*.*

Proof.

uses an extended Erdős-Ko-Rado theorem of Frankl □

Efficiency for well-described $\mathcal{F}$

Theorem (Berstein, Lee, Onn, Weismantel '10)

- *When $\mathcal{F}$ is well described, f is quasi-convex* def*, and W has a fixed number of rows and is unary encoded over binary encoded $\{a_1, \ldots, a_p\}$, we have an efficient deterministic algorithm for maximization*

Efficiency for well-described $\mathcal{F}$

Theorem (Berstein, Lee, Onn, Weismantel '10)

- *When $\mathcal{F}$ is well described, f is quasi-convex* def*, and W has a fixed number of rows and is unary encoded over binary encoded $\{a_1, \ldots, a_p\}$, we have an efficient deterministic algorithm for maximization*
- *When $\mathcal{F}$ is well described, f is a norm, and W is binary-encoded, we have an efficient deterministic constant-approximation algorithm for maximization. (The approx factor depends on the norm, hence on the number of rows of W, while the running time increases only linearly in the number of rows)*

Efficiency for well-described $\mathcal{F}$

Theorem (Berstein, Lee, Onn, Weismantel '10)

- *When $\mathcal{F}$ is well described, f is quasi-convex* ▸ def*, and W has a fixed number of rows and is unary encoded over binary encoded $\{a_1, \ldots, a_p\}$, we have an efficient deterministic algorithm for maximization*
- *When $\mathcal{F}$ is well described, f is a norm, and W is binary-encoded, we have an efficient deterministic constant-approximation algorithm for maximization. (The approx factor depends on the norm, hence on the number of rows of W, while the running time increases only linearly in the number of rows)*
- *When non-negative $\mathcal{F}$ is well described, f is "ray concave"* ▸ def *and non-decreasing, and non-negative W has a fixed number of rows and is unary encoded over binary encoded $\{a_1, \ldots, a_p\}$, we have an efficient deterministic constant-approximation algorithm for minimization.*

Definition

A function $f : \mathbb{R}^d_+ \to \mathbb{R}$ is **ray-concave** if

$$\lambda f(u) \leq f(\lambda u) \text{ for } u \in \mathbb{R}^d_+ \, , \; 0 \leq \lambda \leq 1$$

Definition

A function $f : \mathbb{R}^d_+ \to \mathbb{R}$ is **ray-concave** if

$$\lambda f(u) \leq f(\lambda u) \text{ for } u \in \mathbb{R}^d_+ \;,\; 0 \leq \lambda \leq 1$$

Ordinary concavity of a function f has the special case:

$$\lambda f(u) + (1-\lambda) f(\mathbf{0}) \leq f(\lambda u + (1-\lambda)\mathbf{0}) \;,\; \text{for } u \in \mathbb{R}^d_+ \;,\; 0 \leq \lambda \leq 1 \;,$$

so if f is concave with $f(\mathbf{0}) = 0$, then it is ray-concave.

Definition

A function $f : \mathbb{R}^d_+ \to \mathbb{R}$ is **ray-concave** if

$$\lambda f(u) \le f(\lambda u) \text{ for } u \in \mathbb{R}^d_+ \ , \ 0 \le \lambda \le 1$$

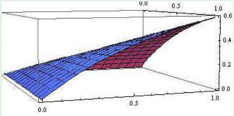
Ordinary concavity of a function f has the special case:

$$\lambda f(u) + (1-\lambda) f(\mathbf{0}) \le f(\lambda u + (1-\lambda)\mathbf{0}) \ , \ \text{for } u \in \mathbb{R}^d_+ \ , \ 0 \le \lambda \le 1 \ ,$$

so if f is concave with $f(\mathbf{0}) = 0$, then it is ray-concave.

Example

- every norm is both ray concave and ray convex on $\mathbb{R}^d_+$.
- $f(u) := \prod_{i=1}^{d} u_i$ is ray convex on $\mathbb{R}^d_+$.
- $f(u) := \min(u_1, u_2)$ is ray concave on $\mathbb{R}^2_+$.



return

Efficiency for well-described independence systems

Theorem (Lee, Onn, Weismantel '09)

For every primitive p-tuple $a = (a_1, \ldots, a_p)$, there is a constant $r(a)$ and an algorithm that, given any well-described independence system $\mathcal{F} \subseteq \{0,1\}^n$, a single weight vector $w \in \{a_1, \ldots, a_p\}^n$, and function $f : \mathbb{Z} \to \mathbb{R}$ presented by a comparison oracle, we give an efficient deterministic algorithm for finding an "$r(a)$-best solution" (to the one-dimensional optimization problem $\max / \min\{f(w'x) \ : \ x \in \mathcal{F}\}$).

Moreover:

- *If a_i divides a_{i+1} for $i = 1, \ldots, p-1$, then the algorithm provides an optimal solution.*
- *For $p = 2$, that is, for $a = (a_1, a_2)$, the algorithm provides an $Fr(a)$-best solution. In particular, for $a = (2,3)$ we give a 1-best solution, while a finding 0-best solution is provably exponential.*

Efficiency for quasi-convex discrete polynomial opt

Theorem (Bank et al. '91,'93)

Let $f, g_1, \ldots, g_m \in \mathbb{Z}[x_1, \ldots, x_n]$ be quasi-convex polynomials of degree at most $d \geq 2$, with coefficients having a binary encoding length of at most ℓ. Let

$$F = \{\, \mathbf{x} \in \mathbb{R}^n : g_i(\mathbf{x}) \leq 0 \quad \text{for } i = 1, \ldots, m \,\}$$

be the (continuous) feasible region. If the integer minimization problem $\min\{\, f(\mathbf{x}) : \mathbf{x} \in F \cap \mathbb{Z}^n \,\}$ is bounded, there exists a radius $R \in \mathbb{Z}_+$ of binary encoding length at most $(md)^{\mathrm{O}(n)}\ell$ such that

$$\min\{\, f(\mathbf{x}) : \mathbf{x} \in F \cap \mathbb{Z}^n \,\} = \min\{\, f(\mathbf{x}) : \mathbf{x} \in F \cap \mathbb{Z}^n,\ \|\mathbf{x}\| \leq R \,\}.$$

Theorem (Heinz '05)

Let $f, g_1, \ldots, g_m \in \mathbb{Z}[x_1, \ldots, x_n]$ be quasi-convex polynomials of degree at most $d \geq 2$, with coefficients having a binary encoding length of at most ℓ. There exists an algorithm running in time $m\ell^{\mathrm{O}(1)} d^{\mathrm{O}(n)} 2^{\mathrm{O}(n^3)}$ that computes a minimizer $\mathbf{x}^* \in \mathbb{Z}^n$ or reports that no minimizer exists.

FPTAS for polynomial optimization in fixed dimension

- An algorithm is a PTAS (for a max problem) if for all fixed $\epsilon > 0$, in polynomial-time the algorithm finds a feasible solution of value within a factor of $(1 - \epsilon)$ of the optimal value
- A FPTAS is a PTAS where the running time is polynomial in $1/\epsilon$

FPTAS for polynomial optimization in fixed dimension

- An algorithm is a PTAS (for a max problem) if for all fixed $\epsilon > 0$, in polynomial-time the algorithm finds a feasible solution of value within a factor of $(1 - \epsilon)$ of the optimal value
- A FPTAS is a PTAS where the running time is polynomial in $1/\epsilon$

Consider

$$\max f(x_1, \ldots, x_n) \ \text{ s.t. } \ x \in P \cap \mathbb{Z}^n, \tag{1}$$

where f is a polynomial of total degree D and $P = \{x | Ax \le b\}$ is a convex polytope, where A is an $m \times n$ integral matrix and b is an integral m-vector.

FPTAS for polynomial optimization in fixed dimension

- An algorithm is a PTAS (for a max problem) if for all fixed $\epsilon > 0$, in polynomial-time the algorithm finds a feasible solution of value within a factor of $(1-\epsilon)$ of the optimal value
- A FPTAS is a PTAS where the running time is polynomial in $1/\epsilon$

Consider

$$\max f(x_1, \ldots, x_n) \ \text{ s.t. } \ x \in P \cap \mathbb{Z}^n, \tag{1}$$

where f is a polynomial of total degree D and $P = \{x | Ax \le b\}$ is a convex polytope, where A is an $m \times n$ integral matrix and b is an integral m-vector.

Theorem (De Loera, Hemmecke, Köppe, Weismantel '06)

Obtain an increasing sequence of lower bounds $\{L_k\}$ and a decreasing sequence of upper bounds $\{U_k\}$ to the optimal value. The bounds can be computed in time polynomial in k , the input size of P and f , and the maximum total degree D and they satisfy the inequality $U_k - L_k \le f^*(\sqrt[k]{|P \cap \mathbb{Z}^n|} - 1)$. More strongly, if f is nonnegative on P, this gives an FPTAS.

Some references

- Raymond Hemmecke, Matthias Köppe, Jon Lee, Robert Weismantel, Nonlinear integer programming. In: M. Jünger, T. Liebling, D. Naddef, G. Nemhauser, W. Pulleyblank, G. Reinelt, G. Rinaldi, and L. Wolsey (eds.), "50 Years of Integer Programming 1958-2008: The Early Years and State-of-the-Art Surveys," Springer-Verlag, 2010, pp. 561–618.
- Shmuel Onn, Nonlinear Discrete Optimization: An Algorithmic Theory. Zürich Lectures in Advanced Mathematics, European Mathematical Society, x+137 pp., September 2010.
- Mixed-Integer Nonlinear Programming and Applications, Jon Lee and Sven Leyffer, eds. IMA Volume 154, 692 pages. Frontier Series, Springer Science + Business Media, LLC, 2012.

III - Non-convex quadratic MINLP

Disjunctive cuts and reformulation for spatial B&B

Some references

- Polynomially-solvable cases of quadratic integer minimization
  - J. Lee, S. Onn, L. Romanchuk, R. Weismantel. The Quadratic Graver Cone, Quadratic Integer Minimization, and Extensions, *Math. Prog., Series B,* 136:301–323, 2012.
- Disjunctive cuts based on dynamically identifying 1-d concavity.
  - A. Saxena, P. Bonami, J. Lee. Convex relaxations of non-convex mixed integer quadratically constrained programs: Extended formulations, *Math. Prog., B,* 124:383–411, 2010.
  - A. Saxena, P. Bonami, J. Lee. Convex relaxations of non-convex mixed integer quadratically constrained programs: Projected formulations, *Math. Prog., A,* 130:359–413, 2011.
- Exploiting separable concavity.
  - C. D'Ambrosio, J. Lee, A. Wačhter. An algorithmic framework for MINLP with separable non-convexity. In: "Mixed Integer Nonlinear Programming", S. Leyffer and J. Lee, Eds., *The IMA Volumes in Mathematics and its Applications,* 154:315–347, 2012.
  - M. Fampa, J. Lee, W. Melo. On global optimization with indefinite quadratics. Preprint, 2014.

III - Non-convex quadratic MINLP

Disjunctive cuts and reformulation for spatial B&B

Secant cut

Our starting point is the equation $X = xx'$. Let $c \in \mathbb{R}^n$ be arbitrary (for now). We have the equation

$$\mathrm{Tr}(X(cc')) = \mathrm{Tr}(xx'(cc')) = (c'x)^2 \ ,$$

which we relax as the concave inequality

$$(c'x)^2 \geq \mathrm{Tr}(X(cc')) \ . \qquad (\Omega)$$

Next, let $\mathcal{P}$ be a relaxation of our feasible region. Let

$$\eta_L(c) := \min \left\{ c'x \ : \ (x, X) \in \mathcal{P} \right\}$$

and

$$\eta_U(c) := \max \left\{ c'x \ : \ (x, X) \in \mathcal{P} \right\} \ .$$

That is, $[\eta_L, \eta_U]$ is the range of $c'x$ as (x, X) varies over the relaxation $\mathcal{P}$. We can convexify to obtain the (linear) secant cut

$$(c'x) \left(\eta_L(c) + \eta_U(c) \right) \ - \ \eta_L(c)\eta_U(c) \geq \mathrm{Tr}(X(cc')) \qquad (SC)$$

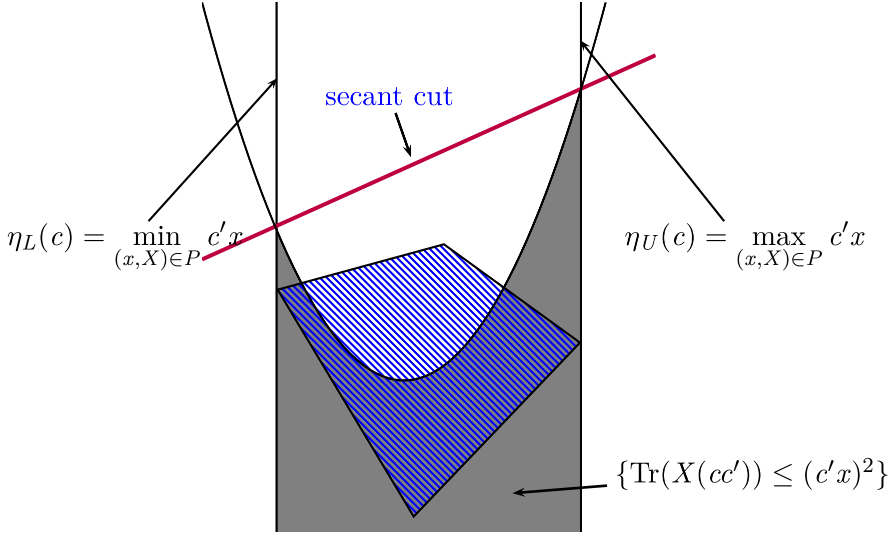


Figure 1: Secant cut from a concave quadratic

A disjunction

Secant inequalities are valid cutting planes, and we may use them. But we can do better, at some computational cost. We choose a value $\theta \in (\eta_L, \eta_U)$ (e.g., the midpoint), and we get the disjunction:

$$\left\{ (x, X) \in \mathcal{P} \ : \ \begin{array}{l} \eta_L(c) \leq c'x \leq \theta \\ (c'x)(\eta_L(c) + \theta) \ - \ \theta\eta_L(c) \geq \mathrm{Tr}(X(cc')) \end{array} \right\}$$

or

$$\left\{ (x, X) \in \mathcal{P} \ : \ \begin{array}{l} \theta \leq c'x \leq \eta_U(c) \\ (c'x)(\eta_U(c) + \theta) \ - \ \theta\eta_U(c) \geq \mathrm{Tr}(X(cc')) \end{array} \right\} .$$

Notice that the second part of the first (resp., second) half of the disjunction corresponds to a secant inequality over the interval between the point θ and the lower (resp., upper) bound for $c'x$.

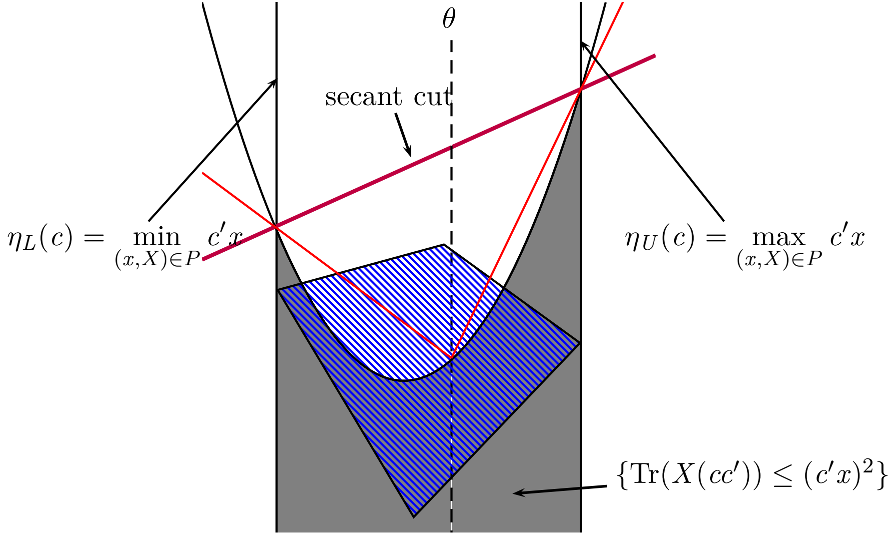


FIGURE 2: A DISJUNCTION OF SECANT CUTS

We have already seen how for <u>any</u> $c \in \mathbb{R}^n$, (Ω) is a valid concave inequality. Now we will see how to use available information to get a <u>violated</u> inequality of this type. If we have a point $(\hat{x}, \hat{X})$ that satisfies the semidefiniteness constraint $X - xx' \succeq 0$, but for which $\hat{X}$ is not equal to $\hat{x}\hat{x}'$, then it is the case that $\hat{X} - \hat{x}\hat{x}'$ has a positive eigenvalue λ . Let c denote a unit-length eigenvector belonging to λ . Then

$$\begin{aligned} \lambda &= \lambda \|c\|_2^2 \\ &= \mathrm{Tr}(\lambda(cc')) \\ &= \mathrm{Tr}((\hat{X} - \hat{x}\hat{x}')(cc')) \ . \end{aligned}$$

So, $\lambda > 0$ if and only if $(c'\hat{x})^2 < \mathrm{Tr}(\hat{X}(cc'))$. That is, every positive eigenvalue of $\hat{X} - \hat{x}\hat{x}'$ yields an inequality of the form (Ω) that is violated by $(\hat{x}, \hat{X})$.
Now, we cannot directly include such a violated inequality, or we would destroy the convexity of our relaxation; so we appeal to a technique of disjunctive programming...

Disjunctive-cut approach

Let's suppose that our relaxation $\mathcal{P}$ is polyhedral. Let $\tilde{x}$ denote a "vectorized" (x, X). So let

$$\mathcal{P} \;:=\; \{\tilde{x} \in \mathbb{R}^N \;:\; A\tilde{x} \geq b\}$$

Next, given our disjunction

$$\mathcal{D}_1 := \{\tilde{x} \in \mathcal{P} \;:\; D_1\tilde{x} \geq d_1\} \text{ or } \mathcal{D}_2 := \{\tilde{x} \in \mathcal{P} \;:\; D_2\tilde{x} \geq d_2\},$$

and a point $\hat{\tilde{x}} \in \mathcal{P}$, our goal is to separate $\hat{\tilde{x}}$ from

$$\mathcal{Q} \;:=\; \operatorname{convcl}(\mathcal{D}_1 \cup \mathcal{D}_2)$$

with a valid linear inequality $\alpha'\tilde{x} \geq \beta$.

The disjunctive cut and the CGLP

$$\begin{aligned} \min\ & \alpha'\hat{\tilde{x}} - \beta \\ & \alpha = A'u_1 + D_1'v_1 \\ & \alpha = A'u_2 + D_2'v_2 \\ & \beta \le b'u_1 + d_1'v_1 \\ & \beta \le b'u_2 + d_2'v_2 \\ & u_1\ ,\ v_1\ ,\ u_2\ ,\ v_2 \ge 0 \\ & (\ \|(u_1, v_1, u_2, v_2)\|_1 = 1\ )\ . \end{aligned} \tag{CGLP}$$

To see that a feasible solution of (CGLP) corresponds to a valid inequality $\alpha'\tilde{x} \ge \beta$, notice that

$$\alpha'\tilde{x} = u_i'A\tilde{x} + v_i'D_i\tilde{x} \ge u_i'b + v_i'd_i \ge \beta\ ,$$

for $i = 1, 2$. A violated linear inequality exists and is found precisely when the minimum of (CGLP) is negative.

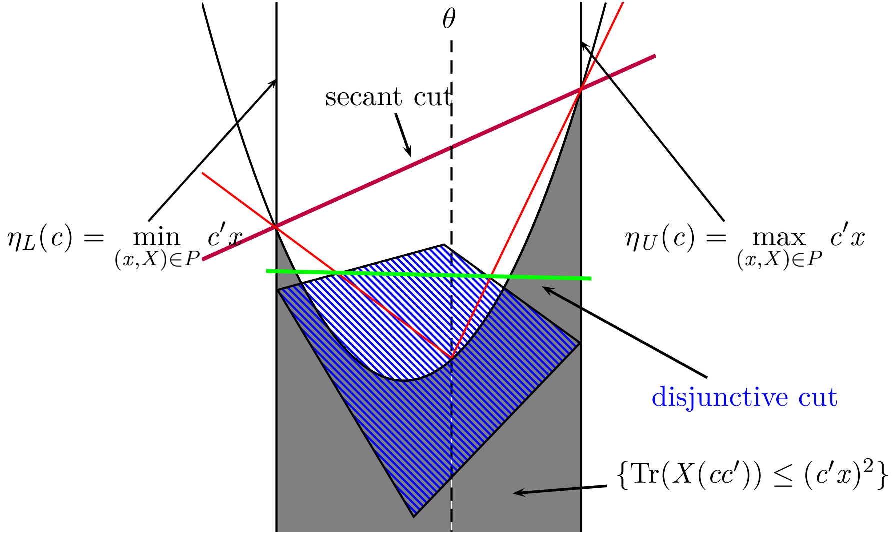


FIGURE 3: DISJUNCTIVE CUT FROM A CONCAVE QUADRATIC

III - Non-convex quadratic MINLP

Disjunctive cuts and **reformulation for spatial B&B**

Our problem:

$$\begin{aligned} z := \min\ & f_0(x) + q_0(x)\ , \\ & f_i(x) + q_i(x) \leq 0\ , i = 1, 2, \ldots, m\ ; \\ & x \in \mathcal{X}\ , \end{aligned}$$

$f_i : \mathbb{R}^n \to \mathbb{R}$ are convex ,
$q_i(x) = \frac{1}{2} x' Q_i x$,
$\mathcal{X}$ is described in a tractable manner by convex functions and possibly integrality restrictions.

But here we focus on:

$$\begin{aligned} z := \min\ & f(x) + q(x)\ , \\ & x \in \mathcal{X}\ . \end{aligned} \tag{I}$$

Main steps of our methodology

- Decompose $q(x) := \frac{1}{2}x'Qx$, into a D.C. (Difference of Convex) quadratic functions:

$$q(x) := \frac{1}{2}x'Px - \frac{1}{2}x'Rx \ , \qquad (P, R \succeq 0).$$

- Underestimate the concave term by a linear function in order to get a convex relaxation of the problem.
- Apply a spatial branch-and-bound algorithm, exploiting convexity of q (extracted into $\frac{1}{2}x'Px$) as much as possible.

Preprocessing via D.C. decompositiion

- Decompose Q:

$$Q = P - R \ , \qquad (P, R \succeq 0).$$

- Calculate the real Schur decomposition of R:

$$R = \sum_{i \in N} \lambda_i v_i v_i' \ , \text{ where } \lambda_i > 0 \text{ for } i \in N \ ,$$

- Define:

$$y_i := \sqrt{\lambda_i} v_i' x \ , \text{ for } i \in N.$$

Problem reformulation, isolating the concavity separably:

$$
\begin{aligned}
z = \min\ & f(x) + \frac{1}{2}x'Px - \frac{1}{2}\sum_{i \in N} y_i^2 \ , \\
& x \in \mathcal{X} \ , \\
& y_i = \sqrt{\lambda_i}\, v_i' x \ , \text{ for } i \in N \ , \\
& l_y \le y \le u_y \ ,
\end{aligned}
\tag{$\widetilde{\mathrm{I}}$}
$$

where

$$
l_{y_i} := \sqrt{\lambda_i} \min_{x \in \mathcal{X}} v_i' x \ , \qquad u_{y_i} := \sqrt{\lambda_i} \max_{x \in \mathcal{X}} v_i' x \ ,
$$

Our spatial branch-and-bound

Let:

$$\omega_i(y_i) := -\tfrac{1}{2} y_i^2 \ ,$$
$$l_{y_i} \leq y_i \leq u_{y_i} \ .$$

The secant under-estimators:

$$-\frac{1}{2}\left((y_i - l_{y_i}) \frac{u_{y_i}^2 - l_{y_i}^2}{u_{y_i} - l_{y_i}} + l_{y_i}^2 \right) \leq w_i \ .$$

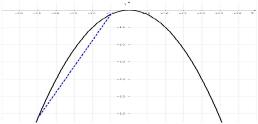


Figure: Secant under-estimator for $-y^2$ in $[-2.5, -0.5]$

Subproblems are relaxations of $\tilde{\mathrm{I}}$

$$
\begin{aligned}
\underline{z} := \min \ & f(x) + \frac{1}{2} x' P x + \sum_{i \in N} w_i \ , \qquad (\underline{\tilde{\mathrm{I}}}) \\
& x \in \mathcal{X} \ , \\
& y_i = \sqrt{\lambda_i} v_i' x \ , \ \text{for } i \in N \ , \\
& -\frac{1}{2} \left((y_i - l_{y_i}) \frac{u_{y_i}^2 - l_{y_i}^2}{u_{y_i} - l_{y_i}} + l_{y_i}^2 \right) \leq w_i \ , \ \text{for } i \in N \ , \\
& l_y \leq y \leq u_y \ .
\end{aligned}
$$

Or equivalently, eliminating y_i:

$$
\begin{aligned}
&\underline{z} = \min\ f(x) + \frac{1}{2} x' P x + \sum_{i \in N} w_i \ , && (\underline{\mathrm{I}})\\
&x \in \mathcal{X}\ ,\\
&-\frac{1}{2}\left(\left(\sqrt{\lambda_i} v_i' x - l_{y_i} \right) \frac{u_{y_i}^2 - l_{y_i}^2}{u_{y_i} - l_{y_i}} + l_{y_i}^2 \right) \le w_i\ ,\ \text{for}\ i \in N\ ,\\
&l_{y_i} \le \sqrt{\lambda_i} v_i' x \le u_{y_i}\ ,\ \text{for}\ i \in N\ .
\end{aligned}
$$

D.C. decomposition strategies

- Our goal: Analyze how different D.C. decompositions affect the global solution of the problem.

D.C. decomposition strategies

- Our goal: Analyze how different D.C. decompositions affect the global solution of the problem.
- Our focus: Minimize the concavity of the D.C. decomposition.

D.C. decomposition strategies

- Our goal: Analyze how different D.C. decompositions affect the global solution of the problem.
- Our focus: Minimize the concavity of the D.C. decomposition.
- Our methodology: Compare different measures for the concavity, that lead to different decompositions of the indefinite matrix Q.

Diagonal decompositions (i.e., R is diagonal)

One advantage: No presence in the subproblems of the dense inequalities

$$l_{y_i} \leq \sqrt{\lambda_i} v_i' x \leq u_{y_i} \ , \ \text{for } i \in N \ .$$

1. Diagonally Dominant

$$r_i := \max\left\{0,\ -q_{ii} + \sum_{j:j\neq i} |q_{ij}|\right\}, \qquad i = 1, 2, \ldots, n.$$

$$R := \mathrm{Diag}(r_1, r_2, \ldots, r_n).$$

$$\Downarrow$$

$P := Q + R$ is diagonally dominant.

2. Identity

$$R := -\min\{0, \lambda_n\}\ I\ ,$$

where λ_n is the least eigenvalue of Q .

$$\Downarrow$$

$$P := Q + R \succeq 0.$$

3. Diagonal SDP

$$\min \sum_{i=1}^{n} r_i \ , \qquad \text{(D-SDP)}$$
$$P := Q + \mathrm{Diag}(r) \succeq 0 \ ,$$
$$r \in \mathbb{R}_+^n \ ,$$

$$\Downarrow$$

$R := \mathrm{Diag}(r)$, the diagonal matrix with minimum trace.

Non-diagonal decompositions (i.e., R may be non-diagonal)

- Hope to capture more convexity into P
- But must deal with dense equations
- And would expect these better decompositions would be more expensive to calculate

1. Real Schur Decomposition

Calculate the Real Schur Decomposition of Q

$$Q = \sum_{i=1}^{n} \lambda_i v_i v_i' \ .$$

Decompose Q

$$Q = P - R \ ,$$

where

$$P := \sum_{i \in P} \lambda_i v_i v_i' \ , \text{ where } \lambda_i > 0 \text{ for } i \in P \ ,$$
$$R := \sum_{i \in N} (-\lambda_i) v_i v_i' \ , \text{ where } \lambda_i < 0 \text{ for } i \in N \ .$$

2. Non-diagonal decompositions via SDP

Let $m_1 \geq m_2 \geq \cdots m_k > 0 =: m_{k+1}$.

$$\begin{aligned} \min \sum_{i=1}^{k} m_i \lambda_i(R) \ , \\ P := Q + R \succeq 0 \ ; \\ R \succeq 0 \ , \end{aligned} \tag{WESDP}$$

- If $k := n$, $m_i := 1$, $\forall i$, $\mathrm{Tr}(R)$ is minimized .
- If $k := 1$, $m_1 := 1$, $\lambda_{\max}(R)$ is minimized.
- If $m_1 >> \cdots >> m_n > 0$, R has a lexically-minimum list of eigenvalues.

Theorem

For any given parameters m_i, $i = 1, \ldots, k$, such that $m_1 \geq m_2 \geq \cdots m_k > 0 =: m_{k+1}$, the splitting determined by the Real Schur Decomposition solves the weighted-eigenvalue minimization problem (WESDP).

Proof idea

Applying a result in [Alizadeh, 1993], we first establish

Lemma

(WESDP) is equivalent to the following dual pair of SDPs:

$$\begin{aligned}
&\min \ \textstyle\sum_{i=1}^{k} i z_i + \sum_{i=1}^{k} Tr(V_i) \ , && \text{(PSDP)}\\
&z_i I + V_i - (m_i - m_{i+1}) R \succeq 0 \ , \quad i = 1, 2, \ldots, k \ ; \\
&Q + R \succeq 0 \ ; \\
&R \succeq 0 \ ; \\
&V_i \succeq 0 \ , \quad i = 1, 2, \ldots, k \ .
\end{aligned}$$

$$\begin{aligned}
&\max \ Q \bullet X \ , && \text{(DSDP)}\\
&Tr(Y_i) = i \ , \quad i = 1, 2, \ldots, k \ ; \\
&\textstyle\sum_{i=1}^{k} (m_i - m_{i+1}) Y_i + X \succeq 0 \ ; \\
&0 \preceq Y_i \preceq I \ , \quad i = 1, 2, \ldots, k \ ; \\
&X \preceq 0 \ .
\end{aligned}$$

Computational experiments: Goals

- Comparison of four different decompositions as preprocessing at the root problem of the spatial branch-and-bound
  - Diagonally Dominant (D-Dom).
  - Identity (Identity).
  - Diagonal SDP (D-SDP).
  - Real Schur Decomposition (RSD).
- Comparison of our methods with Couenne.

Computational experiments: Our setup

- iquad - our C++ coded software.
- Mosek - for solving convex QP relaxations and SDPs for decomposing Q.
- Lapack/Blas - for calculating eigenvalues and Real Schur Decompositions.
- Flux computing cluster at the University of Michigan - processors operating at 2.6 GHz, each can access up to 48GB of RAM (we mostly used no more than 4GB). Each run executed on a single processor.
- Time limit - 2 hours per instance.
- Absolute(relative) convergence tolerance: $10^{-4}(10^{-3})$.

Test problems

- BoxQ - 99 randomly generated box-constrained quadratic programs with Q of various density, $20 \leq n \leq 125$.
- R-BiqMac - 343 problems from the Biq Mac Library, where the integrality constraints are relaxed (box-constrained quadratic programs), $30 \leq n \leq 500$.
- GLOBALLib - 83 problems from GLOBALLib with non-convex quadratic objective function and linear constraints, $2 \leq n \leq 79$.
- Random - 60 randomly generated problems with non-convex quadratic objective function and linear constraints $n = 20, 40, 60, 80, 100$. Number of linear constraints - $1.5n$.

Test-Bed	Time(m)	Splitting Strategy RSD	D-SDP	D-Dom	Identity	Couenne
R-BiqMac	30	0.29	9.04	4.37	2.62	47.23
	60	0.58	11.08	4.37	2.92	48.10
	90	0.87	12.54	4.37	3.50	48.10
	120	0.87	12.83	4.66	3.50	48.40
BoxQP	30	13.13	61.62	9.09	50.51	30.30
	60	14.14	65.66	11.11	51.52	32.32
	90	17.17	65.66	12.12	52.53	33.33
	120	17.17	68.69	14.14	52.53	33.33
GLOBALLib	30	100.00	100.00	97.59	100.00	75.90
	60	100.00	100.00	97.59	100.00	75.90
	90	100.00	100.00	97.59	100.00	75.90
	120	100.00	100.00	97.59	100.00	75.90
Random	30	76.67	23.33	15.00	3.33	48.33
	60	78.33	25.00	18.33	5.00	48.33
	90	83.33	26.67	26.67	5.00	48.33
	120	83.33	28.33	26.67	5.00	48.33

Table: Percentage of problems solved (%)

- R-BiqMac - iquad doesn't succeed for any splitting strategy. Couenne is more successful, although it solves less than 50% of the problems in 2 hs.
- BoxQP - iquad/D-SDP is the best method. Identity is a good alternative: It is better than Couenne and not much worse than D-SDP.
- GLOBALLib - iquad is very good for all splitting strategies, better than Couenne.
- Random - iquad/RSD is the best method.
- Overall the problem categories, D-SDP almost always dominates D-Dom and Identity.
- RSD performs well on instances with linear constraints, being the best alternative on GLOBALLib and Random. RSD doesn't present good results for R-BiqMac and BoxQP. In this case, the diagonal splitting strategies are better.

Test-Bed	Time(m)	Splitting Strategy RSD	D-SDP	D-Dom	Identity
R-BiqMac	30	99.50	1.02	19.27	7.13
	60	99.16	1.00	19.92	7.19
	90	98.84	0.98	20.21	7.21
	120	98.81	0.97	20.59	7.24
BoxQP	30	81.53	0.19	41.03	0.69
	60	78.98	0.17	40.91	0.63
	90	77.98	0.15	40.48	0.60
	120	77.44	0.14	39.87	0.58
GLOBALLib	30	0.00	0.00	2.41	0.00
	60	0.00	0.00	2.41	0.00
	90	0.00	0.00	2.41	0.00
	120	0.00	0.00	2.41	0.00
Random	30	0.03	8.54	26.10	93.33
	60	0.03	8.11	25.06	91.67
	90	0.02	7.89	23.44	91.67
	120	0.02	7.75	23.05	91.67

Table: Normalized gap (%)

- R-BiqMac - D-SDP is the best decomposition strategy, when considering the gap for unsolved problems, and the percentage of problems solved.
- BoxQP - D-SDP is the best decomposition strategy, when considering the gap for unsolved problems, and the percentage of problems solved. Identity also presents good results.
- GLOBALLib - D-Dom is the worst decomposition because of the big gaps left for the small number of unsolved problems.
- Random - RSD wins over all other methods, leaving small gaps for unsolved problems.

Thanks!... Questions?

Definition

The **Frobenius number** is the largest value b for which the Frobenius equation $a_1x_1 + a_2x_2 + \cdots a_px_p = b$ has no solution in non-negative integers.

Definition

The **Frobenius number** is the largest value b for which the Frobenius equation $a_1x_1 + a_2x_2 + \cdots a_px_p = b$ has no solution in non-negative integers.

Definition

The **Frobenius number** is the largest value b for which the Frobenius equation $a_1 x_1 + a_2 x_2 + \cdots a_p x_p = b$ has no solution in non-negative integers.

Coinage as reformed by Augustus c. 23 BCE (1 gold aureus=25 silver denarii; 1 denarius=4 bronze sestertii; 1 sestertius=2 brass dupondii; 1 dupondius=2 copper asses; 1 as=2 bronze semisses; 1 semis=2 copper quadrantes) return

Definition

A function $f : \mathbb{R}^d \to \mathbb{R}$ is **quasi convex** if

$$f(\lambda x + (1 - \lambda)y) \leq \max(f(x), f(y)) \text{ for } x, y \in \mathbb{R}^d \,,\; 0 \leq \lambda \leq 1$$

return

Definition

A function $f : \mathbb{R}^d \to \mathbb{R}$ is **quasi convex** if

$$f(\lambda x + (1-\lambda)y) \leq \max(f(x), f(y)) \text{ for } x, y \in \mathbb{R}^d \ , \ 0 \leq \lambda \leq 1$$

◂ return

Equivalently, the inverse image of any set of the form $(-\infty, a)$ is a convex set. That is, the "lower level sets" are convex.

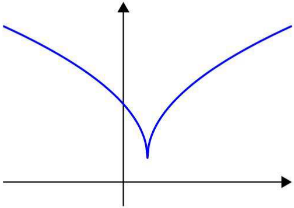
Definition

A function $f : \mathbb{R}^d \to \mathbb{R}$ is **quasi convex** if

$$f(\lambda x + (1-\lambda)y) \leq \max(f(x), f(y)) \text{ for } x, y \in \mathbb{R}^d \ , \ 0 \leq \lambda \leq 1$$

return

Equivalently, the inverse image of any set of the form $(-\infty, a)$ is a convex set. That is, the "lower level sets" are convex.



Problem

Theorem

When $\mathcal{F}$ is well described, f is a norm, and W is binary-encoded, we have an efficient deterministic constant-approximation algorithm for maximization. (The approx factor depends on the norm, hence on the number of rows of W, while the running time increases only linearly in the number of rows). Note: For $1 \leq p \leq \infty$, we get a $d^{\frac{1}{p}}$-approximation.

Problem

Theorem

When $\mathcal{F}$ is well described, f is a norm, and W is binary-encoded, we have an efficient deterministic constant-approximation algorithm for maximization. (The approx factor depends on the norm, hence on the number of rows of W, while the running time increases only linearly in the number of rows). Note: For $1 \leq p \leq \infty$, we get a $d^{\frac{1}{p}}$-approximation.

Prove it!

Problem

Theorem

When $\mathcal{F}$ is well described, f is a norm, and W is binary-encoded, we have an efficient deterministic constant-approximation algorithm for maximization. (The approx factor depends on the norm, hence on the number of rows of W, while the running time increases only linearly in the number of rows). Note: For $1 \le p \le \infty$, we get a $d^{\frac{1}{p}}$-approximation.

Prove it! The Algorithm:

1. For $i = 1, \ldots, d$, solve $\max\{|W_i x^i| \ : \ x^i \in \mathrm{vert}(P)\}$.

Problem

Theorem

When $\mathcal{F}$ is well described, f is a norm, and W is binary-encoded, we have an efficient deterministic constant-approximation algorithm for maximization. (The approx factor depends on the norm, hence on the number of rows of W, while the running time increases only linearly in the number of rows). Note: For $1 \leq p \leq \infty$, we get a $d^{\frac{1}{p}}$-approximation.

Prove it! The Algorithm:

1. For $i = 1, \ldots, d$, solve $\max\{|W_i x^i| \ : \ x^i \in \mathrm{vert}(P)\}$.
2. Let $u^i := W x^i$, for $i = 1, \ldots, d$.

Problem

Theorem

When $\mathcal{F}$ is well described, f is a norm, and W is binary-encoded, we have an efficient deterministic constant-approximation algorithm for maximization. (The approx factor depends on the norm, hence on the number of rows of W, while the running time increases only linearly in the number of rows). Note: For $1 \leq p \leq \infty$, we get a $d^{\frac{1}{p}}$-approximation.

Prove it! The Algorithm:

1. For $i = 1, \ldots, d$, solve $\max\{|W_i x^i| \ : \ x^i \in \mathrm{vert}(P)\}$.
2. Let $u^i := Wx^i$, for $i = 1, \ldots, d$.
3. Output x^{i^*} such that $f(u^{i^*}) = \max_{i=1}^{d} f(u^i)$.

Problem

Lemma

Let $\mathcal{F}$ be well described, and let $\mathcal{P} := \operatorname{conv}(\mathcal{F})$. Let c, $W_{d\times n}$, $\underline{u \in \operatorname{vert}(W\mathcal{P})}$ be binary encoded. Then, in polynomial time, we can solve the "fiber-optimization problem"

$$\max\left\{c'x \;\;:\;\; Wx = u,\;\; x \in \mathcal{P} \cap \mathbb{Z}^n\right\}.$$

Problem

Lemma

Let $\mathcal{F}$ be well described, and let $\mathcal{P} := \mathrm{conv}(\mathcal{F})$. Let c, $W_{d\times n}$, $\underline{u \in \mathrm{vert}(W\mathcal{P})}$ be binary encoded. Then, in polynomial time, we can solve the "fiber-optimization problem"

$$\max\left\{c'x \ : \ Wx = u,\ x \in \mathcal{P} \cap \mathbb{Z}^n\right\}.$$

Prove it! HINT: Think about the preimage of a face of $W\mathcal{P}$.

(How the lemma gets used: when $W_{d\times n}$ has say a unary encoding and d is fixed, we can compute the vertices of $W\mathcal{P}$ efficiently).

Problem

Lemma

Let $\mathcal{F}$ be well described, and let $\mathcal{P} := \mathrm{conv}(\mathcal{F})$. Let c, $W_{d\times n}$, $\underline{u \in \mathrm{vert}(W\mathcal{P})}$ be binary encoded. Then, in polynomial time, we can solve the "fiber-optimization problem"

$$\max\left\{c'x \ : \ Wx = u,\ x \in \mathcal{P} \cap \mathbb{Z}^n\right\}.$$

Prove it! HINT: Think about the preimage of a face of $W\mathcal{P}$.

(How the lemma gets used: when $W_{d\times n}$ has say a unary encoding and d is fixed, we can compute the vertices of $W\mathcal{P}$ efficiently).