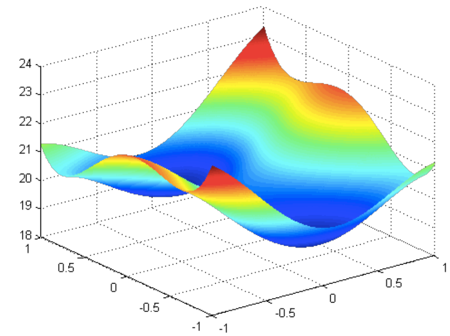
Across the Watershed: Unifying Perspectives in Optimization

Pablo A. Parrilo
Laboratory for Information and Decision Systems
Electrical Engineering and Computer Science
Massachusetts Institute of Technology

www.mit.edu/~parrilo

Optimization?



- Goes back to many classical ideas in mathematics and physics (Fermat, Newton, Euler…)
- Variational principles (least-action, Hamilton, Euler-Lagrange, calculus of variations, etc)
- More recently, convexity and complexity (Farkas, Minkowski, Caratheodory, Kantorovich, Dantzig, Khachiyan, Karmarkar…)
- Strong links with game theory, control theory, combinatorics, TCS, etc…

Dido's problem
(Virgil's *Aeneid*)

Brachistochrone
(Bernoulli, 1696)

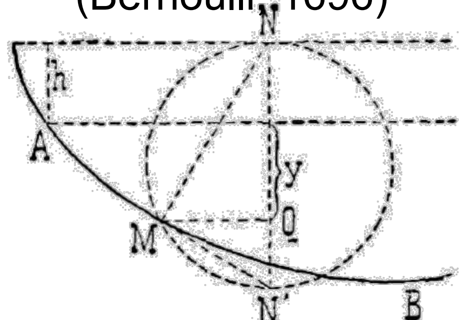


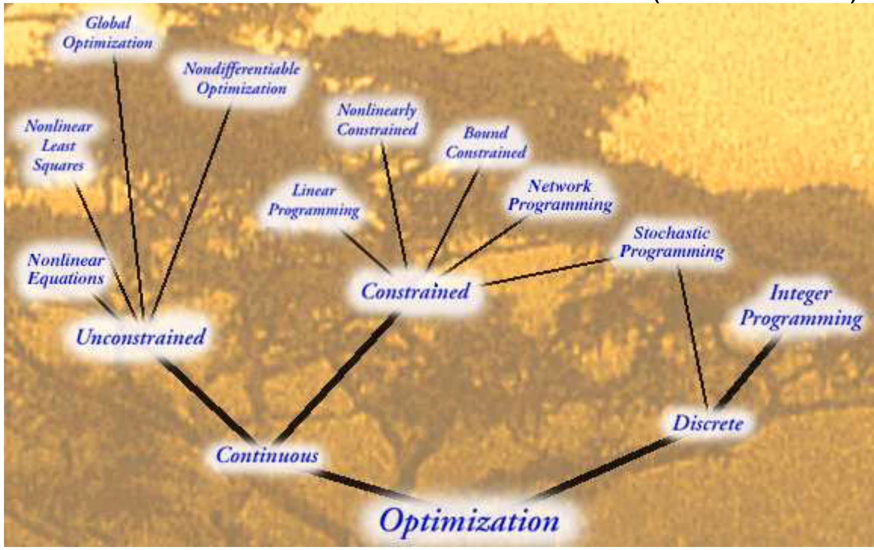
Fig. 2.

Optimization is ubiquitous

- Optimization is essential across many scientific and engineering applications (signal processing, robotics, VLSI, machine learning, mechanical design, revenue management, ...)
- Often, defines what an "acceptable solution" is
- Enables whole industries:
  - Airlines: jet engine design, CFD, route planning, fare pricing, crew/plane scheduling, maintenance, …
  - Finance and insurance: trading, derivatives, statistical modeling and optimization, ...
  - E-commerce: combinatorial auctions, ad campaign design, recommendation systems, …

(NEOS server)

Many flavors



Demand for increasingly sophisticated mathematical optimization methods:

- From 1950s on: linear programming, nonlinear, global, convex, quadratic, semidefinite, hyperbolic, etc.
- Combinatorial, network, packing/covering, integer, submodular, etc.

Mathematical infrastructure and associated computational methods for engineering and scientific applications.

The convexity watershed

"...in fact, the great watershed in optimization isn't between linearity and nonlinearity, but convexity and nonconvexity." (p. 185)

SIAM Review
Vol. 35, No. 2, pp. 183–238, June 1993

001

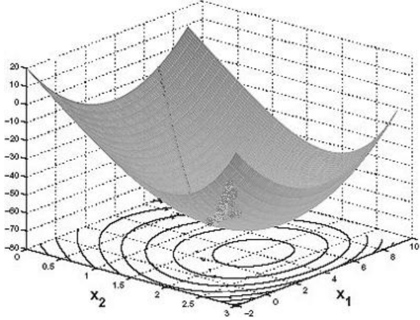
LAGRANGE MULTIPLIERS AND OPTIMALITY*

R. TYRRELL ROCKAFELLAR†

Abstract. Lagrange multipliers used to be viewed as auxiliary variables introduced in a problem of constrained minimization in order to write first-order optimality conditions formally as a system of equations. Modern applications, with their emphasis on numerical methods and more complicated side conditions than equations, have demanded deeper understanding of the concept and how it fits into a larger theoretical picture.

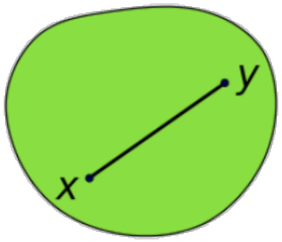
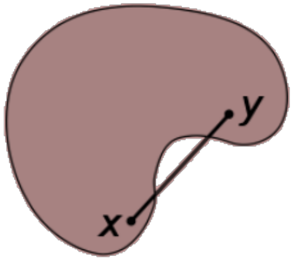
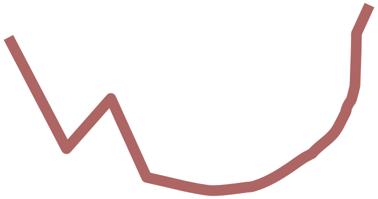
A major line of research has been the nonsmooth geometry of one-sided tangent and normal vectors to the set of points satisfying the given constraints. Another has been the game-theoretic role of multiplier vectors

Convexity

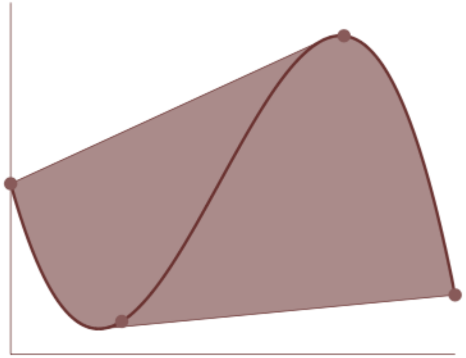


Sets: $x, y \in S \quad \Rightarrow \quad \lambda x + (1-\lambda) y \in S \quad \forall \lambda \in [0,1]$

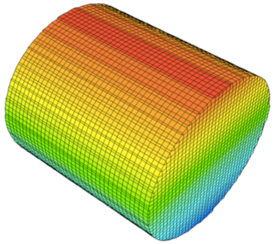
Functions: $f(\lambda x + (1-\lambda) y) \leq \lambda f(x) + (1-\lambda) f(y)$

	Convex	**Non-convex**
Sets		
Functions		

Why Convexity?

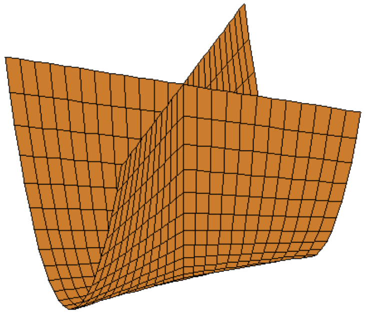
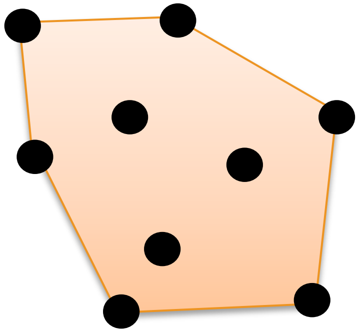
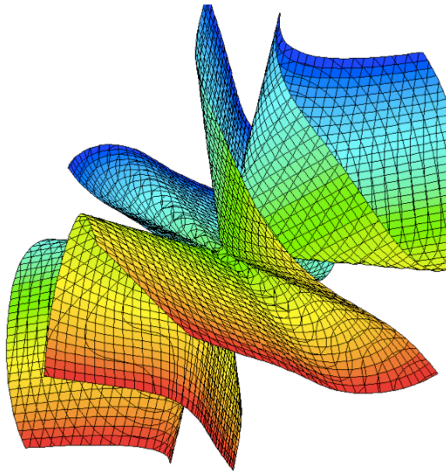
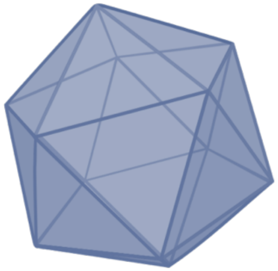


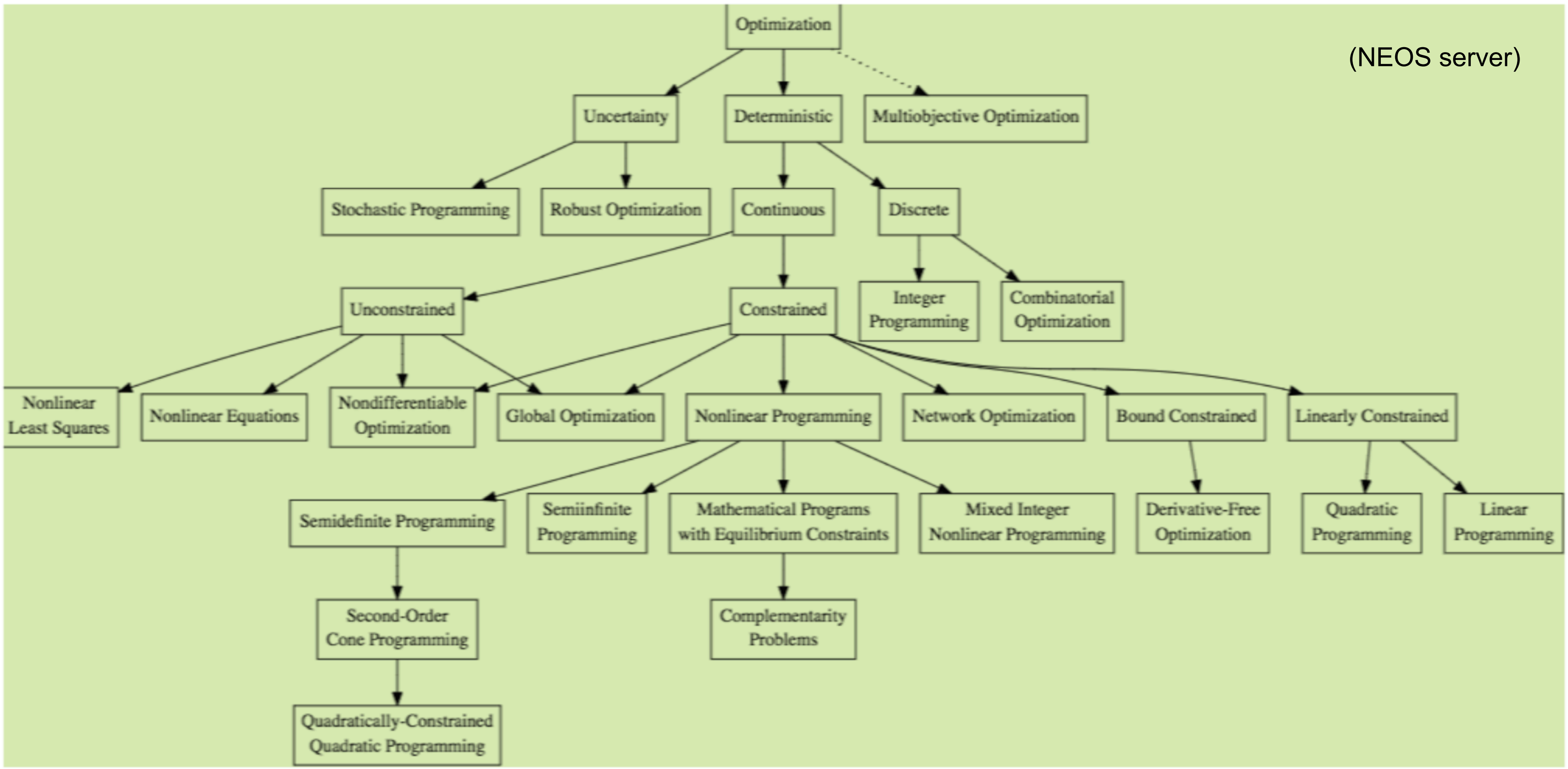
- Simple, but rich, geometric structure
- Principled, modular modeling approach (e.g., Boyd's "disciplined convex programming")
- Predictable algorithmic behavior
- Efficient in theory (polynomial time)
- Remarkably effective in practice
- Many successful applications



Actually, many watersheds…

- Convex vs. Nonconvex
- Linear vs. Nonlinear
- Continuous vs. Discrete
- Constrained vs. Unconstrained
- …


Optimization
Uncertainty
Deterministic
Multiobjective Optimization
Stochastic Programming
Robust Optimization
Continuous
Discrete
Integer Programming
Combinatorial Optimization
Unconstrained
Constrained
Nonlinear Least Squares
Nonlinear Equations
Nondifferentiable Optimization
Global Optimization
Nonlinear Programming
Network Optimization
Bound Constrained
Linearly Constrained
Semidefinite Programming
Semiinfinite Programming
Mathematical Programs with Equilibrium Constraints
Mixed Integer Nonlinear Programming
Derivative-Free Optimization
Quadratic Programming
Linear Programming
Second-Order Cone Programming
Complementarity Problems
Quadratically-Constrained Quadratic Programming
(NEOS server)

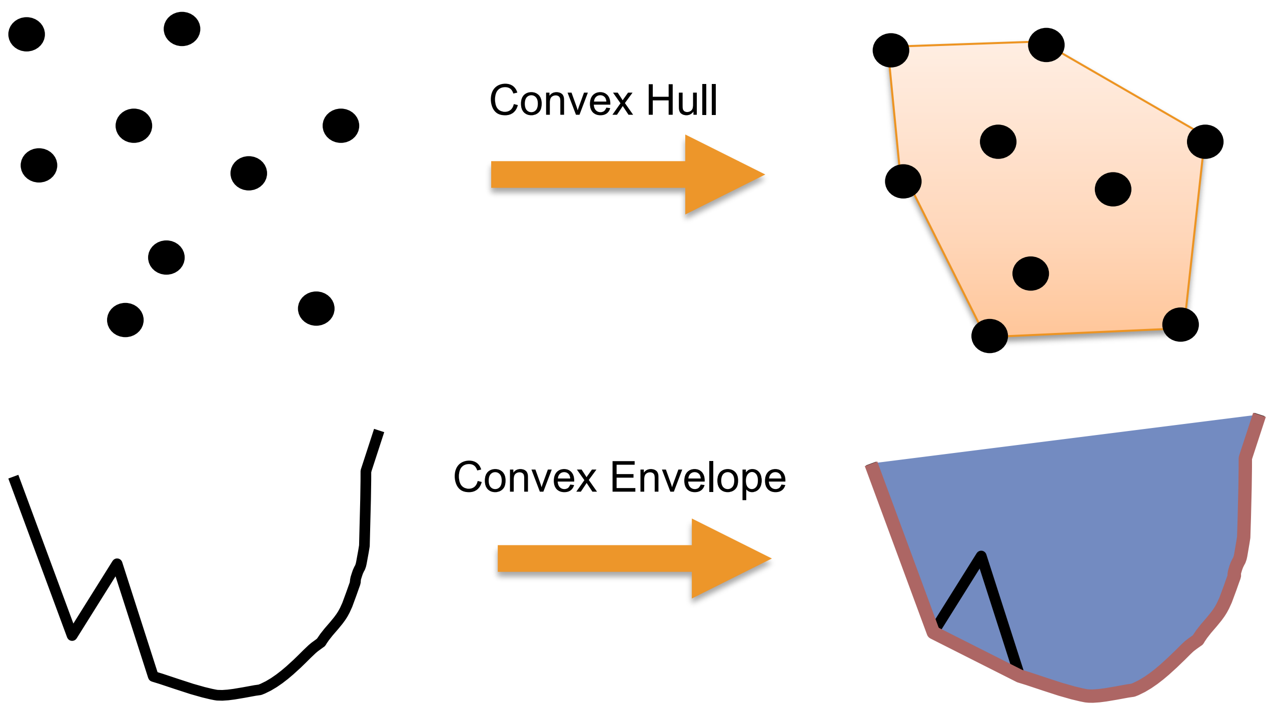
Actually, many watersheds…

All important, and relevant in suitable contexts.

- Still, how much of optimization can be done while bridging these distinctions?
- How “real” are they? Are they really sharp?
- What can we gain (if anything) from a unified perspective?

Analogies: linearity/curvature, invertibility/condition numbers, complexity/parameterized complexity, etc.

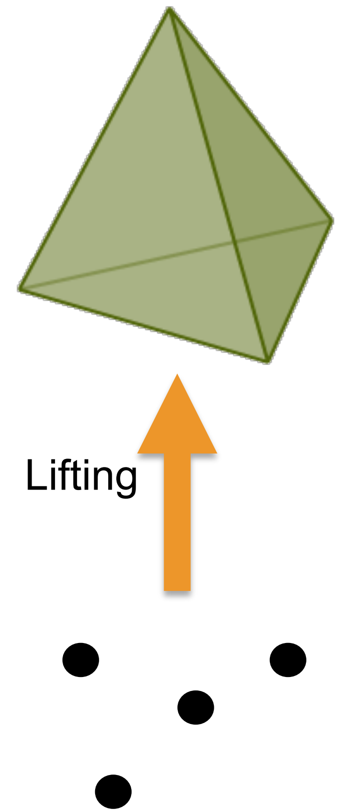
Make things convex?



Very useful, but

- Descriptions may be hard to obtain
- May lose information about points in interior

Convexity is *relative*



- Every set/problem can be "lifted" to a convex setting (in general, infinite dimensional).

- Conceptually easy: a new dimension for each point in the set!

  - E.g., n-point set -> n-dimensional simplex
  - "continuous" set -> infinite-dimensional space
  - *Not* the same thing as taking convex hull (but related)
  - Nice projection of extreme points

- Many examples/interpretations: probability distributions over a set, functional spaces, mixed strategies in games, "relaxed" controls, etc.

- Related theme: in physics/control, same for linear vs. non-linear: dynamics can always be linearized, e.g., Liouville equation).

So, is everything convex?

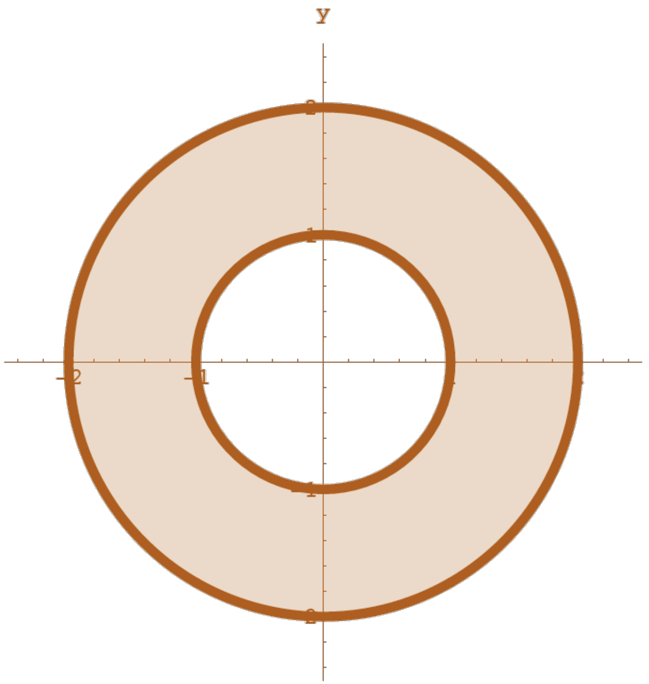
Yeah, but…

Great idea, but often not very practical (as such)

Q: Perhaps we can get away with **finite** (or small) dimension??

Nice. Do you have anything smaller?

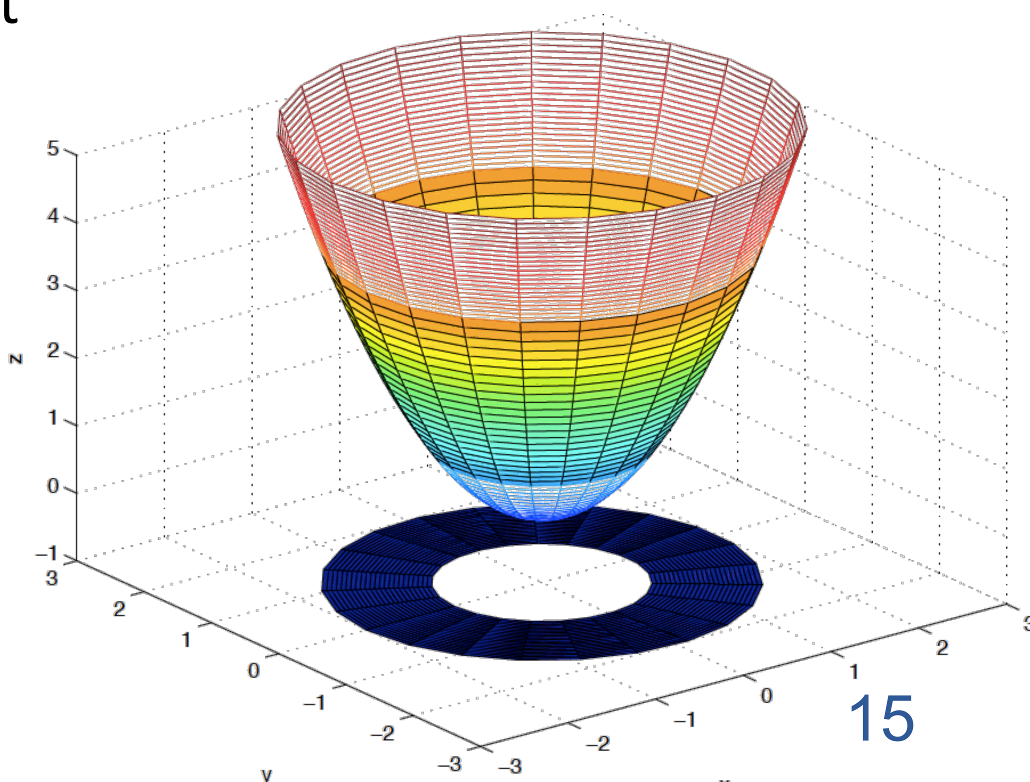
- Interestingly, however, often a **finite** (and small) dimension may be enough.
- **Ex**: Consider the set defined by $1 \leq x^2 + y^2 \leq 2$
- Obviously non-convex.
- Can we use convex optimization?



Small liftings (sometimes) work!

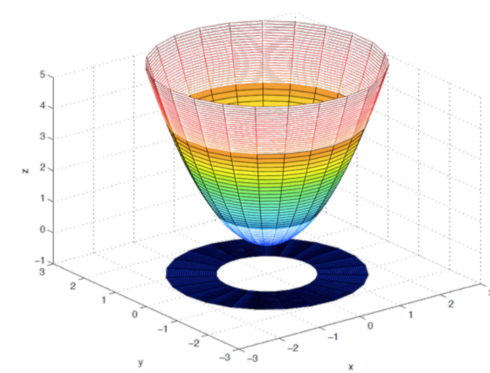
- A polynomial "lifting" to a higher dimensional space: $(x,y) \rightarrow (x,y,x^2+y^2)$
- The original nonconvex set is the <u>projection</u> of the extreme points of a convex set.
- In particular, the convex set defined by

$$x^2 + y^2 \leq z, \qquad 1 \leq z \leq 4$$

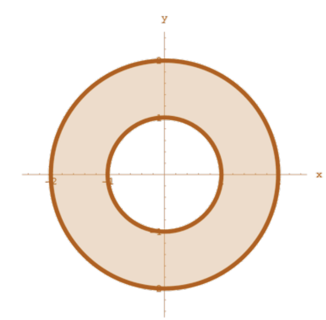


Ascent towards convexity…

Lifted set
(infinite dimensional)



Finite-dimensional
(small) liftings



Original (nonconvex) set

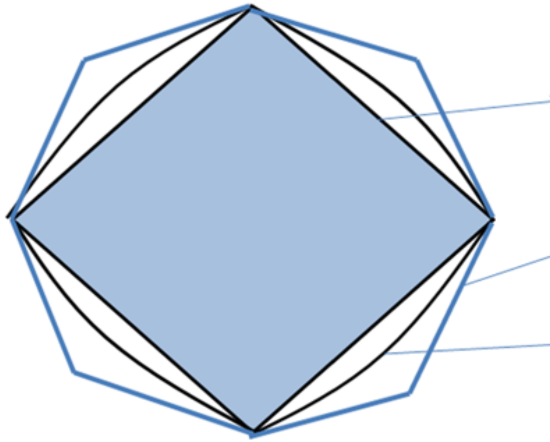
Want to understand and develop *systematic* and *efficient* lifting methods

Answer: **Hierarchies of relaxations**

Hierarchies? What's that?

Long history in optimization:

- Integer programming (Chvatal-Gomory)
- Roof duality (Boros-Crama-Hammer)
- Reformulation/linearization (Sherali-Adams)
- Lift-project / matrix cuts (Lovasz-Schrijver)
- Sum of squares (Shor, P., Lasserre)



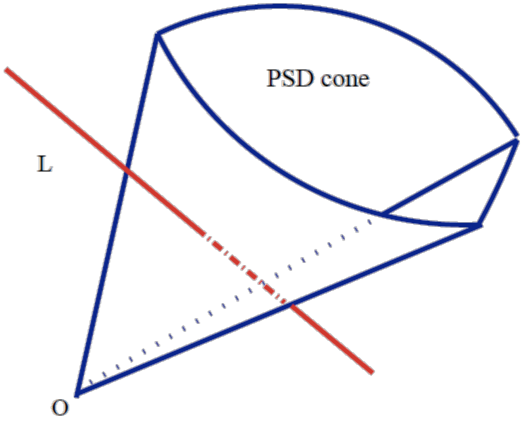
Applicable to general optimization problems described by polynomial equations and constraints.

Yields nice convex problems: semidefinite programs (SDPs).

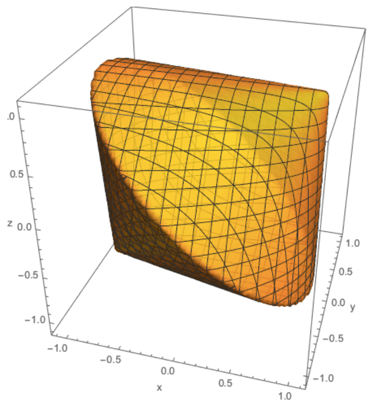
Semidefinite programming (SDP)

A broad generalization of LP to symmetric matrices

$$\min \operatorname{Tr} CX \qquad \text{s.t.} \quad X \in \mathcal{L} \cap \mathcal{S}_+^n$$



- Intersection of affine subspace and cone of positive semidefinite matrices
- Lots of applications
- Originated in combinatorial optimization and control theory. Nowadays, used everywhere.
- Convex, finite dimensional. Nice duality theory
- Not polyhedral, solvable in polynomial time



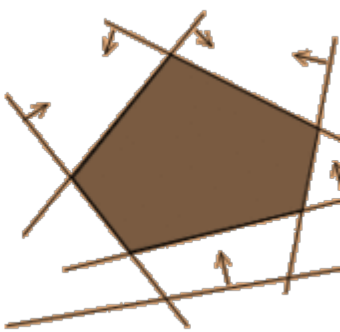
Hierarchies of relaxations

Parameterized families of maps, with two properties:

a) Map each point in the base space, to a point "upstairs"
   E.g., the Veronese embedding

$$\phi : V \to \mathrm{Sym}^k(V), \qquad x \mapsto \underbrace{x \otimes x \otimes \cdots \otimes x}_{k \text{ times}}$$

b) But (crucially!) also must be able to effectively describe or approximate the *convex hull of the image*.



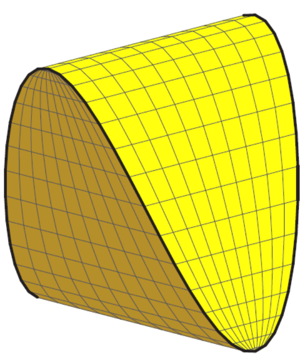
How? Linear functions on Sym^k are polynomials!

$$\ell(\phi(x)) = \ell(x \otimes \cdots \otimes x) =: p(x)$$

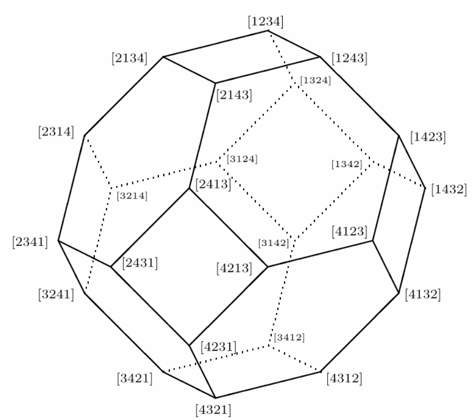
Need to understand **polynomial nonnegativity**…

Convex hulls of real varieties

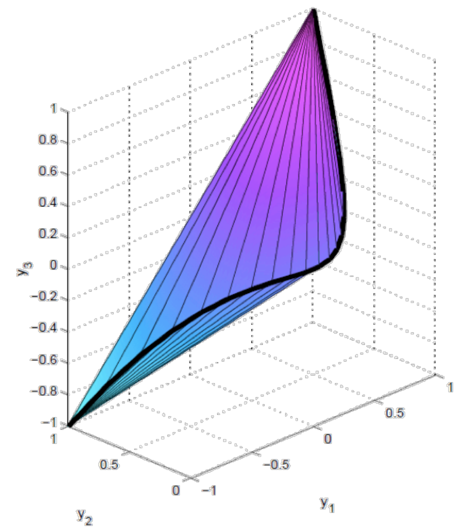
- Need to "effectively" understand convex hulls
- Many levels:
  - Geometrically (e.g., facial structure)
  - Algebraically (e.g., degrees, equations)
  - Computationally (e.g., SDP relaxations)
- Classical question in combinatorial optimization, but continuous aspect adds difficulties



Sanyal-Sottile-Sturmfels, "Orbitopes"
Ranestad-Sturmfels, "Convex hull of a variety"



Permutahedron



Gouveia-Laurent-P.-Thomas, "Theta bodies"

Sums of squares (SOS)

Simplest "nonlinear" problem: polynomial nonnegativity

$$p(x) \geq 0 \quad \forall x \in R^n$$

An "obvious" sufficient condition: SOS

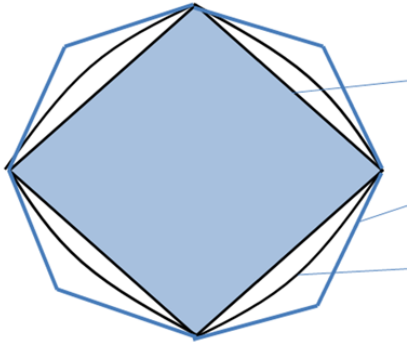
Trivial !

$$p(x) = \sum_{i=1}^{m} q_i^2(x)$$

Surprisingly powerful, and computable via SDP.

Lots of consequences…

Back to hierarchies!



Map each point "upstairs" via

$$\phi : V \to \mathrm{Sym}^k(V), \qquad x \mapsto \underbrace{x \otimes x \otimes \cdots \otimes x}_{k \text{ times}}$$

Replace convex hull with "SOS-convex hull", where we only look at inequalities for which

$$p_\ell(x) = \ell(\phi(x)) \text{ is SOS}$$

- Progressively better conditions for increasing *k*
- For every fixed *k*, polytime solvable, since $\dim(\mathrm{Sym}^k) = O(n^k)$
- Gives a complete hierarchy, as *k* -> infinity
  (under mild assumptions, details omitted…)

Ascent towards convexity…

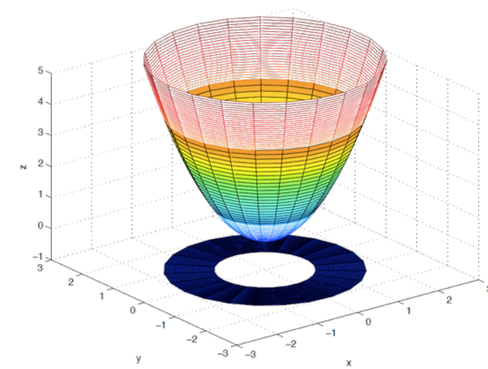
Hierarchies give us an explicit way of trading off **convexity** vs. **dimension**

Tradeoffs are quantifiable, may depend on specific problem class.

What makes a hierarchy "good"? Quantitative and empirical results. Even very "bad" hierarchies may converge asymptotically.

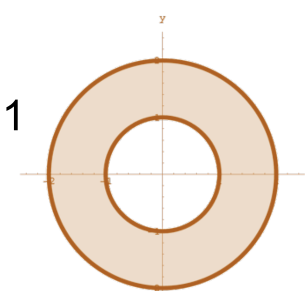
k=inf

Lifted set
(infinite dimensional)



Finite-dimensional
(small) liftings

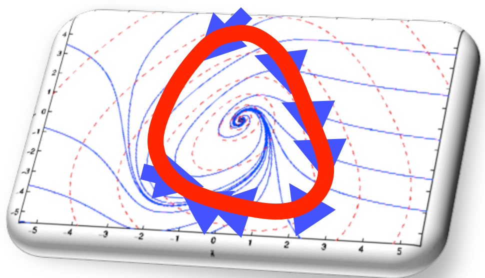
k=1



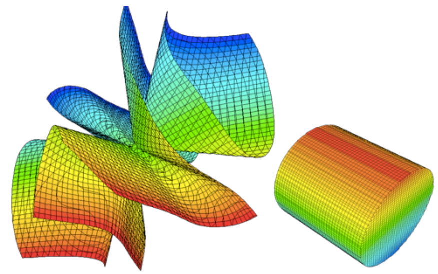
Original (nonconvex) set

Many applications!

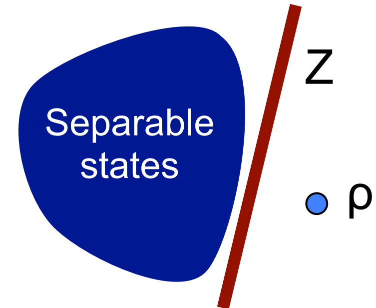
(essentially, anywhere polynomials appear)



Dynamical systems and control theory

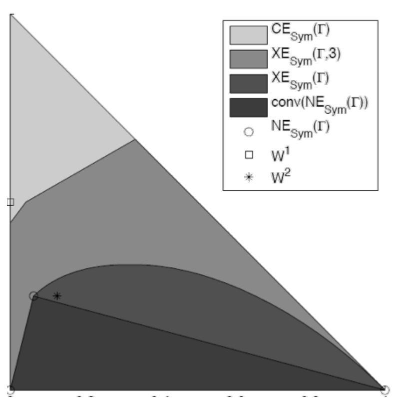


Rank minimization, nuclear norm, compressed sensing

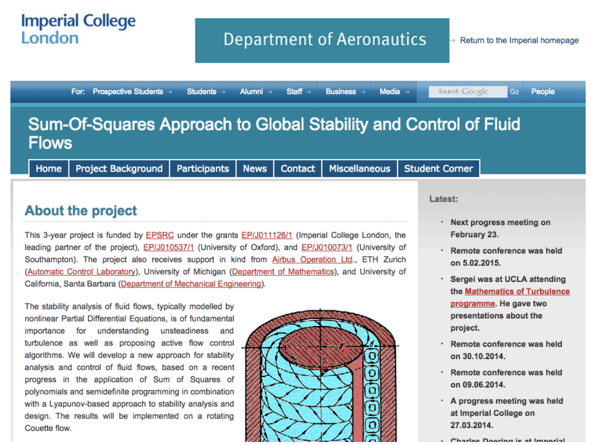


$$\rho = \sum p_i \sigma_i \otimes \omega_i, 0 \leq p_i, \sum p_i = 1$$

Quantum information and entanglement



Game equilibria: computation and refinements



Imperial College London

Department of Aeronautics

Return to the Imperial homepage

For: Prospective Students | Students | Alumni | Staff | Business | Media | Search Google | Go | People

Sum-Of-Squares Approach to Global Stability and Control of Fluid Flows

Home | Project Background | Participants | News | Contact | Miscellaneous | Student Corner

About the project

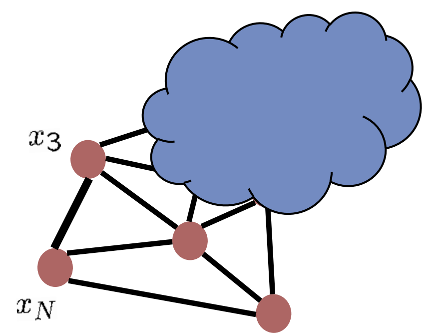
This 3-year project is funded by EPSRC under the grants EP/J011126/1 (Imperial College London, the leading partner of the project), EP/J010537/1 (University of Oxford), and EP/J010073/1 (University of Southampton). The project also receives support in kind from Airbus Operation Ltd., ETH Zurich (Automatic Control Laboratory), University of Michigan (Department of Mathematics), and University of California, Santa Barbara (Department of Mechanical Engineering).

The stability analysis of fluid flows, typically modelled by nonlinear Partial Differential Equations, is of fundamental importance for understanding unsteadiness and turbulence as well as proposing active flow control algorithms. We will develop a new approach for stability analysis and control of fluid flows, based on a recent progress in the application of Sum of Squares of polynomials and semidefinite programming in combination with a Lyapunov-based approach to stability analysis and design. The results will be implemented on a rotating Couette flow.

Latest:

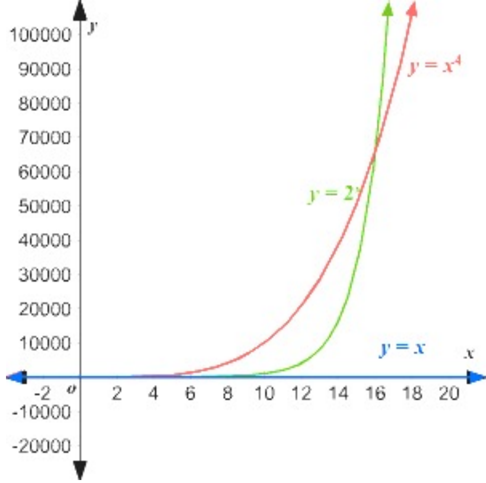
- Next progress meeting on February 23.
- Remote conference was held on 5.02.2015.
- Sergei was at UCLA attending the Mathematics of Turbulence programme. He gave two presentations about the project.
- Remote conference was held on 30.10.2014.
- Remote conference was held on 09.06.2014.
- A progress meeting was held at Imperial College on 27.03.2014.

Fluid dynamics



Latent-variable graphical model selection

Theory & Practice



Very strong relaxations
(but, won't solve NP-hard problems!)
Most powerful known general-purpose technique.

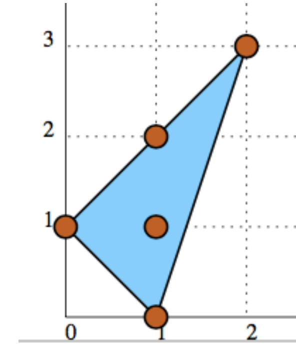
But, recall SDP size is $O(n^d)$

Unless *d* or *n* are "small", may be difficult to solve in practice.

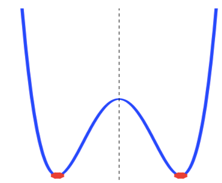
- For generic continuous problems, extremely competitive if global solutions are needed
- For problem classes where certification of solutions is required, essentially unmatched
- For purely discrete problems, not too useful so far (enumeration is way too cheap!)
- Nevertheless, among best asymptotic methods…

But, we can do much better if we exploit structure …

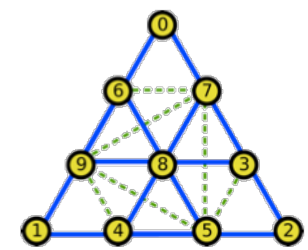
Key: Exploiting structure



- Algebraic structure:
  - Sparsity, Newton polytopes, facial reduction.
  - Ideal structure, SOS on quotient/coordinate rings.
  - Graphical: dependency graph, bounded treewidth.
  - Symmetries: group invariance, representation theory



- Numerical structure:
  - Interpolation and rank-one SDPs (e.g., SDPT3)
  - Displacement rank, fast solvers
  - Sums/intersections of easier cones (chordality, S/DSOS, etc).



Math Connections

- Probability theory (moments, exchangeability, de Finetti, etc)
- Real algebraic geometry (Positivstellensatz)
- Operator theory (via Gelfand-Neimark-Segal)
- Quantum information (separability, entanglement)
- Harmonic analysis on semigroups
- Noncommutative algebra/probability (NC-SOS)
- Complexity and proof theory (certificate degree)
- Graphs/combinatorics (perfect graphs, graphons, flag algebras)
- Tropical geometry (SDP over more general fields)
- …

Current research directions

Upcoming workshop (Nov. 6-9) @Simons !

What can these do (or not do)? Analysis / lower bounds:

- Barak/Brandao/Harrow/Kelner/Steurer/Zhou (SOS(d) solves all known "hard instances" of Unique Games).
- Barak/Hopkins/Kelner/Kothari/Moitra/Potechin, Deshpande/Montanari, etc. (SOS(o(log n)) cannot do better than $n^{(1/2-o(1))}$ for planted clique)
- Lee/Raghavendra/Steurer (e.g., more general relaxations are no better, and no poly-sized SDP can beat 7/8 for MAX3SAT)

Even stronger relaxations?

- Ultimately, need novel ways of certifying inequalities
- Bienstock/Zuckerberg? (not quite automatizable/implementable)
- Fawzi/Saunderson/P. (in restricted class of problems, can do exponentially better than SOS)

Practical challenges

Upcoming workshop (Nov. 6-9) @Simons !

Wonderful when it works, but SDPs quickly get big!

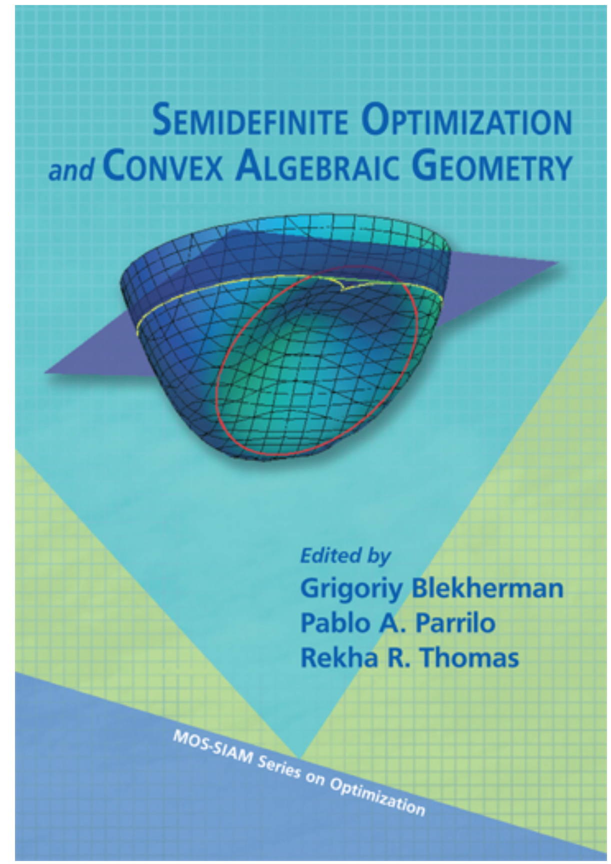
Intrinsic efficiency barriers
(e.g. linear vs. nonlinear approximation theory)

Scalability: even if convex, poly-space is too large!
(alternatives? e.g., low-rank Burer-Monteiro, fast spectral algorithms – Hopkins/Schramm/Shi/Steurer)

Algorithms that are efficient in practice,
not just in Asymptopia.

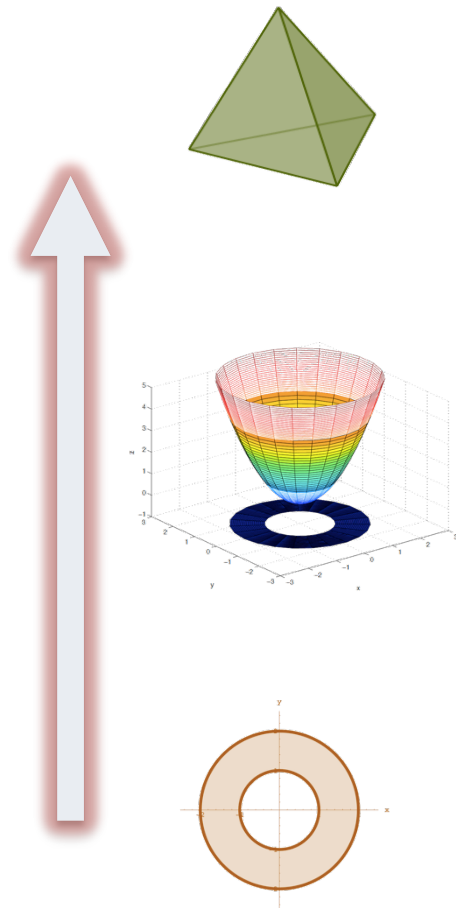
Convex Algebraic Geometry

- ***Optimization +***
  Convex Geometry +
  Algebraic Geometry
- ***Convex*** *sets,* ***algebraic*** *structure*
- Pervasive role of **duality**
- Exploit this structure to develop convex optimization solutions, with global properties



G. Blekherman, P. Parrilo, R. Thomas, ``*Semidefinite Optimization and Convex Algebraic Geometry*,” MOS-SIAM Optimization Series, 2013.
www.mit.edu/~parrilo/sdocag/

Summary



- Convex+Algebraic methods surprisingly powerful
- “Backwards compatible,” nicely generalize earlier successful techniques
- Remarkably effective for small and large problems
- Right at the boundary of known theoretically efficient methods
- Exploiting structure is fundamental in practice

Thanks for your attention!