DTU Summer School on Modern Optimization in Energy Systems

Conic Optimization: Part II

Pascal Van Hentenryck

Goals of the Lecture

- Conic Optimization
  - Semi-Definite Programming

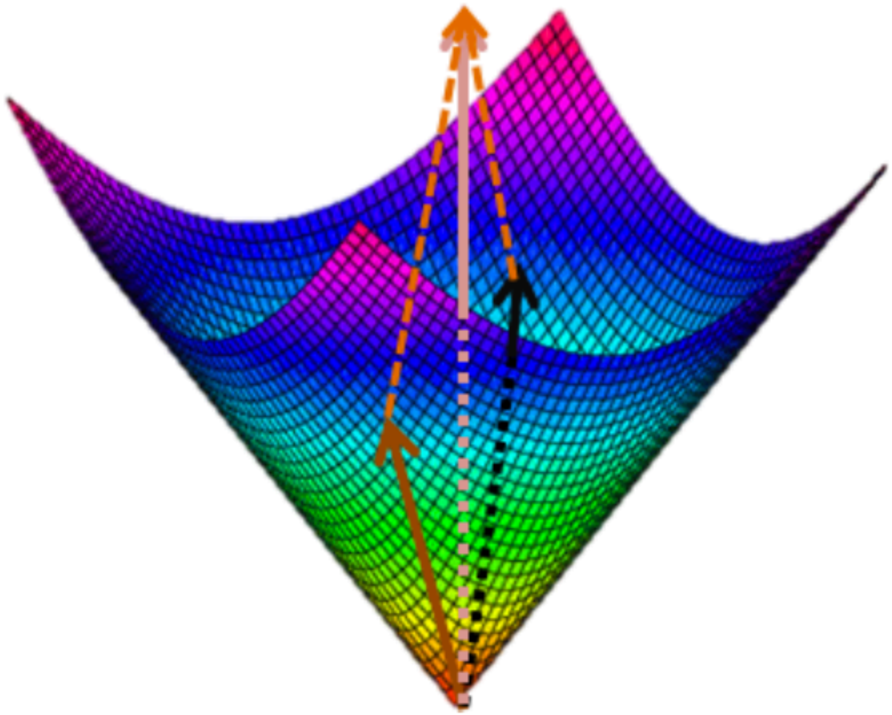
Motivation

- What is really annoying from a convexity standpoint?

$$xy$$

- Key property of a product
  - it is commutative

Beautiful Cones



Notations

- Vectors $x \in \Re^n$
- Symmetric Matrices: $M \in S_n$ with $a_{ij} = a_{ji} \qquad (1 \leq i, j \leq n)$
- Product $x^T y = \sum_{i=1}^{n} x_i y_i$
- Quadratic form

$$x^T M y = \sum_{i=1}^{n} \sum_{j=1}^{n} m_{ij} x_i y_j$$

Positive Semi-Definite Matrix

- Eigenvalues

$$Ax = \lambda x$$

- Property: If M is symmetric, then all its eigenvalues are real
- A symmetric matrix M is positive semi-definite if all its eigenvalues are nonnegative
- Equivalences
  - M is positive semidefinite
  - $\forall x \in \Re^n : x^T M x \geq 0$
  - There exists a matrix $U \in \Re^{n \times n} : M = U^T U$

Positive Semi-Definite Matrix

- M is positive semidefinite

$$M \succeq 0$$

Traces

Traces

- The trace of a matrix is the sum of its diagonal elements

$$Tr(A) = \sum_{i=1}^{n} a_{ii}$$

- The scalar product of matrices

$$\langle X, Y \rangle = Tr(X^T Y) = \sum_{i=1}^{n} \sum_{j=1}^{n} x_{ij} y_{ij}$$

Traces

$$\begin{pmatrix} x_{11} & x_{21} \\ x_{12} & x_{22} \end{pmatrix} \begin{pmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{pmatrix} = \begin{pmatrix} x_{11}y_{11} + x_{21}y_{21} & \\ & x_{12}y_{12} + x_{22}y_{22} \end{pmatrix}$$

Semi-Definite Programming (SDP)

$$\begin{aligned} \min_{X \in S_n} \quad & Tr(CX) \\ s.t. \quad & Tr(A_i X) = b_i \qquad 1 \leq i \leq m \\ & X \succeq 0 \end{aligned}$$

- key insight
  - optimization over matrices

Semi-Definite Programming (SDP)

$$
\begin{aligned}
\min_{X \in S_n} \quad & Tr(CX) \\
s.t. \quad & AX = b_i \\
& X \succeq 0
\end{aligned}
$$

$$
\begin{pmatrix} x_{11} & \dots & x_{1n} \\ & \dots & \\ x_{n1} & \dots & x_{nn} \end{pmatrix} \in S_n \qquad \begin{pmatrix} c_{11} & \dots & c_{1n} \\ & \dots & \\ c_{n1} & \dots & c_{nn} \end{pmatrix} \in S_n
$$

Semidefinite Programming

INSANITY™

Max Cut

- Graph: G = (V,E)
- Goal
  - partition V into two sets, S and V \ S to maximize

$$\sum_{i \in S} \sum_{j \in V \setminus S} I((i,j) \in E) + \sum_{i \in V \setminus S} \sum_{j \in S} I((i,j) \in E)$$

Nonlinear model for Max-Cut

- Decision variables $x_i \in \{-1, 1\}$

- Cut: $(S, V \setminus S)$ where $S = \{i \in V | x_i = 1\}$

- Notations: $w_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$

- Objective $\frac{1}{2} \sum_{i=1}^{n} \sum_{j=i+1}^{n} w_{ij}(1 - x_i x_j)$

- Note that (i,j) is a cut if and only if $x_i x_j = -1$

MaxCut Nonlinear Model

$$\max \quad \sum_{i<j} w_{ij}(1 - x_i x_j)$$

$$s.t. \quad x_i \in \{-1, 1\}$$

MaxCut Nonlinear Model

$$\max \quad \sum_{i<j} w_{ij}(1 - x_i x_j)$$

$$s.t. \quad x_i^2 = 1$$

SDP Relaxation

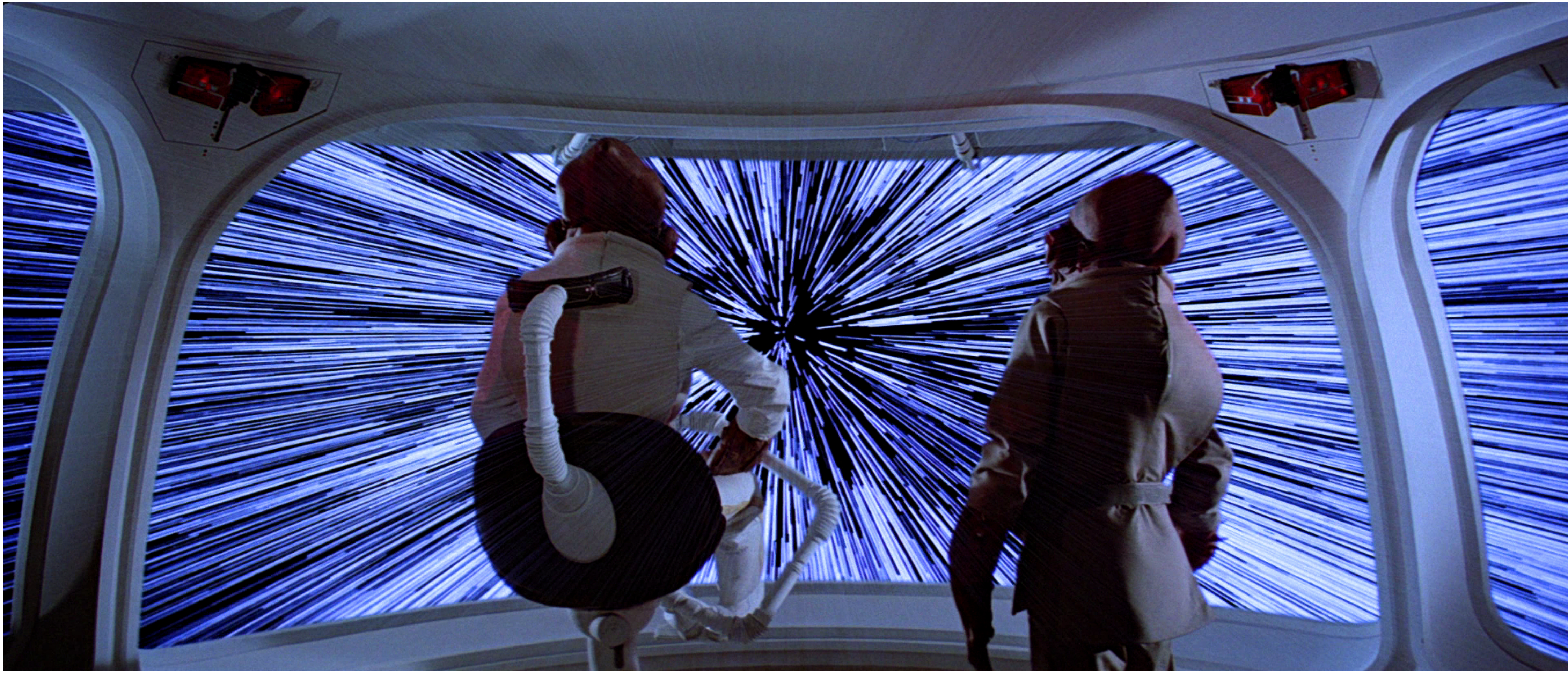
- Define a matrix

$$x_{ij} = x_i x_j$$

- Solve the problem

$$\begin{aligned} \max \quad & \sum_{i<j} w_{ij}(1 - x_{ij}) \\ s.t. \quad & x_{ii} = 1 \\ & X \succeq 0 \end{aligned}$$

Hyperspace



SDP Relaxation

- Finding a SDP relaxation of MaxCut
  - replace x_i by a n-dimensional vector v_i
- The problem now becomes

$$\begin{aligned} \max \quad & \sum_{i<j} w_{ij}(1 - v_i^T v_j) \\ s.t. \quad & \|v_i\|^2 = 1 \end{aligned}$$

- Why is this a relaxation?

SDP Relaxation

$$\max \quad \sum_{i<j} w_{ij}(1 - v_i^T v_j)$$

$$s.t. \quad \|v_i\|^2 = 1$$

- Why is this a relaxation?

$$l : \Re \to \Re^n : x \mapsto (x, 0, \ldots, 0)$$

- if $(x_1, \ldots, x_n)$ is a solution to MaxCut, then

$$\sum_{i=1}^{n} \sum_{j=i+1}^{n} w_{ij}(1 - l(x_i)l(x_j)) = \sum_{i=1}^{n} \sum_{j=i+1}^{n} w_{ij}(1 - x_i x_j)$$

SDP Relaxation

- Finding a SDP relaxation of MaxCut
  - replace xi by a n-dimensional vector vi
- The problem now becomes

$$
\begin{aligned}
\max \quad & \sum_{i<j} w_{ij}(1 - v_i^T v_j) \\
s.t. \quad & \|v_i\|^2 = 1
\end{aligned}
$$

- Why is this a relaxation?
  - there are more solutions (the feasible space is larger)

Magic

SDP Relaxation

- We still do not have a SDP relaxation

- Main construction: introduce product variables
  - define $v_{ij} = v_i^T v_j$

- Optimization model

$$
\begin{array}{ll}
\max & \displaystyle\sum_{i<j} w_{ij}(1 - v_{ij}) \\
s.t. & v_{ii} = 1
\end{array}
$$

- This is an amazingly poor relaxation!

SDP Relaxation

- Create a matrix

$$U = \begin{array}{c} \begin{array}{ccc} v_1 & \qquad & v_n \end{array} \\ \begin{pmatrix} v_{11} & \dots & v_{1n} \\ & \dots & \\ v_{n1} & \dots & v_{nn} \end{pmatrix} \end{array}$$

SDP Relaxation

- Define a matrix

$$V = U^T U$$

- By definition, V is symmetric and positive semi-definite

- Its elements are

$$v_{ij} = v_i^T v_j$$

- So I know that the matrix V of products is positive semi-definite

SDP Relaxation

- The SDP Relaxation

$$\begin{aligned} \max \quad & \sum_{i<j} w_{ij}(1 - v_{ij}) \\ s.t. \quad & v_{ii} = 1 \\ & V \succeq 0 \end{aligned}$$

- This is an amazing relaxation
  - randomized rounding gives a solution no worse than 0.878 times the optimum in the worst case

Summary of Conic Relaxation

- Conic Program

$$\begin{aligned} \min \quad & c^T x \\ s.t. \quad & Ax = b \\ & x \in \mathcal{K} \end{aligned}$$

- Cones
  - the non-negative orthant: $\Re^n_+$
  - the second-order cone: $Q^n = \{(x,t) \in \Re^n_+ : \|x\| \leq t\}$
  - the semi-definite cone: $S^n_+ = \{X : X \succeq t\}$