15.093 Optimization Methods

Lecture 24: Semidefinite Optimization
1 Outline

1. SDO formulation
2. The Maximum cut problem
3. Minimizing Polynomials as an SDP
4. Linear Difference Equations and Stabilization
5. Barrier Algorithm for SDO

2 SDO formulation

2.1 Primal and dual

\[ (P) : \min C \bullet X \]
\[ \text{s.t. } A_i \bullet X = b_i \quad i = 1, \ldots, m \]
\[ X \succeq 0 \]

\[ (D) : \max \sum_{i=1}^{m} y_i b_i \]
\[ \text{s.t. } C - \sum_{i=1}^{m} y_i A_i \succeq 0 \]

3 MAXCUT

- Given \( G = (N, E) \) undirected graph, weights \( w_{ij} \geq 0 \) on edge \((i, j) \in E\)
- Find a subset \( S \subseteq N \): \( \sum_{i \in S, j \in S} w_{ij} \) is maximized
- \( x_j = 1 \) for \( j \in S \) and \( x_j = -1 \) for \( j \in \bar{S} \)

\[ \text{MAXCUT} : \max \frac{1}{4} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} (1 - x_i x_j) \]
\[ \text{s.t. } x_j \in \{-1, 1\}, \quad j = 1, \ldots, n \]

3.1 Reformulation

- Let \( Y = xx' \), i.e., \( Y_{ij} = x_i x_j \)
- Let \( W = [w_{ij}] \)
• Equivalent Formulation

\[
\text{MAXCUT} : \quad \max \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} - W \cdot Y \\
\text{s.t. } x_j \in \{-1, 1\}, \ j = 1, \ldots, n \\
\quad Y_{jj} = 1, \ j = 1, \ldots, n \\
\quad Y = xx'
\]

3.2 Relaxation

• \( Y = xx' \geq 0 \)

• Relaxation

\[
\text{RELAX} : \quad \max \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} - W \cdot Y \\
\text{s.t. } Y_{jj} = 1, \ j = 1, \ldots, n \\
\quad Y \succeq 0
\]

3.3 Feasible set

• For \( n = 3 \), we have

\[
\begin{bmatrix}
1 & Y_{12} & Y_{13} \\
Y_{12} & 1 & Y_{23} \\
Y_{13} & Y_{23} & 1
\end{bmatrix} \succeq 0
\]

An outer approximation to the true feasible set.

• \( \text{MAXCUT} \leq \text{RELAX} \)

• It turns out that:

\[
0.87856 \ \text{RELAX} \leq \text{MAXCUT} \leq \text{RELAX}
\]

• The value of the SDO relaxation is guaranteed to be no more than 12% higher than the value of the very difficult to solve problem MAXCUT
4 Minimizing Polynomials

4.1 Sum of squares

- A polynomial \( f(x) \) is a sum of squares (SOS) if
  \[ f(x) = \sum_j g_j^2(x) \]
  for some polynomials \( g_j(x) \).
- A polynomial satisfies \( f(x) \geq 0 \) for all \( x \in \mathbb{R} \) if and only if it is a sum of squares.
- **Not** true in more than one variable!

4.2 Proof

\( \Rightarrow \) Obvious. If \( f(x) = \sum_j g_j^2(x) \) then \( f(x) \geq 0 \).

\( \Rightarrow \) Factorize \( f(x) = C \prod_j (x-r_j)^{n_j} \prod_k (x-a_k+ib_k)^{m_k} \). \( x-a_k-ib_k \). Since \( f(x) \) is nonnegative, then \( C \geq 0 \) and all the \( n_j \) are even. Then,

\[
\begin{align*}
f_1(x) &= C \frac{1}{2} \prod_j (x-r_j)^{n_j} \prod_k (x-a_k)^{m_k} \quad \text{and} \\
f_2(x) &= C \frac{1}{2} \prod_j (x-r_j)^{n_j} \prod_k b_k^{m_k} 
\end{align*}
\]

4.3 SOS and SDO

- Let \( \tilde{x} = (1, x, x^2, \ldots, x^k)' \).
- \( f(x) = \tilde{x}'Q\tilde{x} \) is a sum of squares if and only if
  \[ f(x) = \tilde{x}'Q\tilde{x}, \]
  where \( Q \succeq 0 \), i.e., \( Q = L'L \).
- Then, \( f(x) = \tilde{x}'L'\tilde{x} = ||L\tilde{x}||^2 \).

4.4 Formulation

- Consider \( \min f(x) \).
- Then, \( f(x) \geq \gamma \) if and only if \( f(x) - \gamma = \tilde{x}'Q\tilde{x} \) with \( Q \succeq 0 \). This implies linear constraints on \( \gamma \) and \( Q \).
- Reformulation

\[
\begin{align*}
\max \gamma \\
\text{s.t.} \quad \begin{cases} 
    f(x) - \gamma &= \tilde{x}'Q\tilde{x} \\
    Q &\succeq 0
\end{cases}
\end{align*}
\]
4.5 Example

4.5.1 Reformulation

\[
\begin{align*}
\min f(x) &= 3 + 4x + 2x^2 + 2x^3 + x^4. \\
f(x) - \gamma &= 3 - \gamma + 4x + 2x^2 + 2x^3 + x^4 = (1, x, x^2)'Q(1, x, x^2).
\end{align*}
\]

\[
\begin{align*}
\max \gamma \\
\text{s.t.} \quad &3 - \gamma = q_{11} \\
&4 = 2q_{12}, \quad 2 = 2q_{13} + q_{22} \\
&2 = 2q_{23}, \quad 1 = q_{33} \\
Q &= \begin{bmatrix} q_{11} & q_{12} & q_{13} \\
q_{12} & q_{22} & q_{23} \\
q_{13} & q_{23} & q_{33} \end{bmatrix} \geq 0
\end{align*}
\]

Extensions to multiple dimensions.

5 Stability

- A linear difference equation
  \[
  x(k + 1) = Ax(k), \quad x(0) = x_0
  \]

- \( x(k) \) converges to zero if \( \lambda_i(A) \) is less than 1, \( i = 1, \ldots, n \)

- Characterization:
  \[
  |\lambda_i(A)| < 1 \quad \forall i \iff \exists P > 0 \quad A'PA - P < 0
  \]

5.1 Proof

- \((\Leftarrow)\) Let \( Av = \lambda v \). Then,
  \[
  0 > v'(A'PA - P)v = (|\lambda|^2 - 1)v'Pv,
  \]
  and therefore \( |\lambda| < 1 \)

- \((\Rightarrow)\) Let \( P = \sum_{i=0}^{\infty} A'^iQA^i \), where \( Q > 0 \). The sum converges by the eigenvalue assumption. Then,
  \[
  A'PA - P = \sum_{i=1}^{\infty} A'^iQA^i - \sum_{i=0}^{\infty} A'^iQA^i = -Q < 0
  \]
5.2 Stabilization

- Consider now the case where $A$ is not stable, but we can change some elements, e.g., $A(L) = A + LC$, where $C$ is a fixed matrix.
- Want to find an $L$ such that $A + LC$ is stable.
- Use Schur complements to rewrite the condition:

$$
\begin{bmatrix}
P & (A + LC)' P \\
P(A + LC) & P
\end{bmatrix} \succ 0
$$

Condition is nonlinear in $(P, L)$

5.3 Changing variables

- Define a new variable $Y := PL$

$$
\begin{bmatrix}
P & A' P + C' Y' \\
PA + YC & P
\end{bmatrix} \succ 0
$$

- This is linear in $(P, Y)$.
- Solve using SDO, recover $L$ via $L = P^{-1} Y$

6 Primal Barrier Algorithm for SDO

- $X \succeq 0 \iff \lambda_1(X) \geq 0, \ldots, \lambda_n(X) \geq 0$
- Natural barrier to repel $X$ from the boundary $\lambda_1(X) > 0, \ldots, \lambda_n(X) > 0$:

$$
- \sum_{j=1}^{n} \log(\lambda_j(X)) =
$$

$$
- \log(\prod_{j=1}^{n} \lambda_j(X)) = - \log(\det(X))
$$

- Logarithmic barrier problem

$$
\begin{aligned}
\min_{X} & \quad B_\mu(X) = C \bullet X - \mu \log(\det(X)) \\
\text{s.t.} & \quad A_i \bullet X = b_i, \quad i = 1, \ldots, m, \\
& \quad X \succeq 0
\end{aligned}
$$

- Derivative: $\nabla B_\mu(X) = C - \mu X^{-1}$

Follows from

$$
\log \det(X + H) \approx \log \det(X) + \text{trace}(X^{-1} H) + \cdots
$$
• KKT conditions

\[ A_i \cdot X = b_i, \quad i = 1, \ldots, m, \]
\[ C - \mu X^{-1} = \sum_{i=1}^{m} y_i A_i. \]
\[ X \succ 0, \]

• Given \( \mu \), need to solve these nonlinear equations for \( X, C, y_i \)
• Apply Newton’s method until we are “close” to the optimal
• Reduce value of \( \mu \), and iterate until the duality gap is small

6.1 Another interpretation

• Recall the optimality conditions:

\[ A_i \cdot X = b_i, \quad i = 1, \ldots, m, \]
\[ \sum_{i=1}^{m} y_i A_i + S = C \]
\[ X, S \succeq 0, \]
\[ XS = 0 \]

• Cannot solve directly. Rather, perturb the complementarity condition to \( XS = \mu I \).
• Now, unique solution for every \( \mu > 0 \) (the “central path”)
• Solve using Newton, for decreasing values of \( \mu \).

7 Differences with LO

• Many different ways to linearize the nonlinear complementarity condition

\[ XS = \mu I \]

• Want to preserve symmetry of the iterates
• Several search directions, most common is Nesterov-Todd.

8 Convergence

8.1 Stopping criterion

• The point \((X, y_i)\) is feasible, and has duality gap:

\[ C \cdot X - \sum_{i=1}^{m} y_i b_i = \mu X^{-1} \cdot X = n \mu \]

• Therefore, reducing \( \mu \) always decreases the duality gap
• Barrier algorithm needs \( O(\sqrt{n} \log \frac{\epsilon_0}{\epsilon}) \) iterations to reduce duality gap from \( \epsilon_0 \) to \( \epsilon \)
9 Conclusions

- SDO is a very powerful modeling tool
- SDO represents the present and future in continuous optimization
- Barrier and primal-dual algorithms are very powerful
- Many good solvers available: SeDuMi, SDPT3, SDPA, etc.
- Pointers to literature and solvers:
  www-user.tu-chemnitz.de/~helmberg/semidef.html