Lagrangian Relaxations and Randomization

- Semidefinite relaxations
- Lagrangian relaxations for general QCQPs
- Randomization
- Bounds on suboptimality (MAXCUT)
Easy & Hard Problems

classical view on complexity:

- **linear** is easy

- **nonlinear** is hard(er)
... EE364 (and correct) view:

- **convex** is easy

- **nonconvex** is hard(er)
Convex Optimization

minimize \( f_0(x) \)
subject to \( f_1(x) \leq 0, \ldots, f_m(x) \leq 0 \)

\( x \in \mathbb{R}^n \) is optimization variable; \( f_i : \mathbb{R}^n \rightarrow \mathbb{R} \) are convex:

\[
f_i(\lambda x + (1 - \lambda)y) \leq \lambda f_i(x) + (1 - \lambda)f_i(y)
\]
for all \( x, y, 0 \leq \lambda \leq 1 \)

- includes LS, LP, QP, and many others
- like LS, LP, and QP, convex problems are fundamentally tractable
Nonconvex Problems

nonconvexity makes problems essentially untractable...

- sometimes the result of bad problem formulation

- however, often arises because of some natural limitation: fixed transaction costs, binary communications, ...
Example: Robust LP

minimize \( c^T x \)
subject to \( \text{Prob}(a_i^T x \leq b_i) \geq \eta, \quad i = 1, \ldots, m \)

coefficient vectors \( a_i \) IID, \( \mathcal{N}(\bar{a}_i, \Sigma_i) \); \( \eta \) is required reliability

- for fixed \( x \), \( a_i^T x \) is \( \mathcal{N}(\bar{a}_i^T x, x^T \Sigma_i x) \)
- so for \( \eta = 50\% \), robust LP reduces to LP

\[
\begin{align*}
\text{minimize} \quad & c^T x \\
\text{subject to} \quad & \bar{a}_i^T x \leq b_i, \quad i = 1, \ldots, m
\end{align*}
\]

and so is easily solved

- what about other values of \( \eta \), e.g., \( \eta = 10\% ? \eta = 90\% ? \)
Hint

\{ x \mid \text{Prob}(a_i^T x \leq b_i) \geq \eta, \ i = 1, \ldots, m \}
That’s right

robust LP with reliability $\eta = 90\%$ is convex, and **very easily solved**

robust LP with reliability $\eta = 10\%$ is not convex, and **extremely difficult**

moral: **very difficult** and **very easy** problems can look **quite similar** (to the untrained eye)
Nonconvex Problems

what can be done?... we will use convex optimization results to:

- find bounds on the optimal value, by **relaxation**
- get "good" feasible points via **randomization**
Basic Problem

- focus here on a specific class of problems: general QCQPs
- vast range of applications...

the generic QCQP can be written:

\[
\begin{align*}
\text{minimize} & \quad x^T P_0 x + q_0^T x + r_0 \\
\text{subject to} & \quad x^T P_i x + q_i^T x + r_i \leq 0, \quad i = 1, \ldots, m
\end{align*}
\]

- if all \( P_i \) are p.s.d., convex problem, use EE364...
- here, we suppose at least one \( P_i \) not p.s.d.
Example: Boolean Least Squares

Boolean least-squares problem:

\[
\begin{align*}
\text{minimize} & \quad \|Ax - b\|^2 \\
\text{subject to} & \quad x_i^2 = 1, \quad i = 1, \ldots, n
\end{align*}
\]

- basic problem in digital communications
- could check all \(2^n\) possible values of \(x\) . . .
- an NP-hard problem, and very hard in practice
- many heuristics for approximate solution
Example: Partitioning Problem

two-way partitioning problem described in §5.1.4 of the 364 reader:

\[
\begin{align*}
\text{minimize} & \quad x^T W x \\
\text{subject to} & \quad x_i^2 = 1, \quad i = 1, \ldots, n
\end{align*}
\]

where \( W \in S^n \), with \( W_{ii} = 0 \).

• a feasible \( x \) corresponds to the partition

\[
\{1, \ldots, n\} = \{i \mid x_i = -1\} \cup \{i \mid x_i = 1\}
\]

• the matrix coefficient \( W_{ij} \) can be interpreted as the cost of having the elements \( i \) and \( j \) in the same partition.

• the objective is to find the partition with least total cost

• classic particular instance: MAXCUT (\( W_{ij} \geq 0 \))
Convex Relaxation

the original QCQP:

\[
\begin{align*}
\text{minimize} & \quad x^T P_0 x + q_0^T x + r_0 \\
\text{subject to} & \quad x^T P_i x + q_i^T x + r_i \leq 0, \quad i = 1, \ldots, m
\end{align*}
\]

can be rewritten:

\[
\begin{align*}
\text{minimize} & \quad \text{Tr}(X P_0) + q_0^T x + r_0 \\
\text{subject to} & \quad \text{Tr}(X P_i) + q_i^T x + r_i \leq 0, \quad i = 1, \ldots, m \\
& \quad X \succeq xx^T \\
& \quad \text{Rank}(X) = 1
\end{align*}
\]

the only nonconvex constraint is now \( \text{Rank}(X) = 1 \ldots \)
Convex Relaxation: Semidefinite Relaxation

- we can directly relax this last constraint, i.e. drop the nonconvex \( \text{Rank}(X) = 1 \) to keep only \( X \succeq xx^T \)

- the resulting program gives a lower bound on the optimal value

\[
\begin{align*}
\text{minimize} & \quad \text{Tr}(XP_0) + q_0^T x + r_0 \\
\text{subject to} & \quad \text{Tr}(XP_i) + q_i^T x + r_i \leq 0, \quad i = 1, \ldots, m \\
& \quad X \succeq xx^T
\end{align*}
\]

Tricky. . . Can be improved?
Lagrangian Relaxation

from the original problem:

\[
\begin{align*}
\text{minimize} & \quad x^T P_0 x + q_0^T x + r_0 \\
\text{subject to} & \quad x^T P_i x + q_i^T x + r_i \leq 0, \quad i = 1, \ldots, m
\end{align*}
\]

we can form the Lagrangian:

\[
L(x, \lambda) = x^T \left( P_0 + \sum_{i=1}^m \lambda_i P_i \right) x + \left( q_0 + \sum_{i=1}^m \lambda_i q_i \right)^T x + r_0 + \sum_{i=1}^m \lambda_i r_i
\]

in the variables \( x \in \mathbb{R}^n \) and \( \lambda \in \mathbb{R}^m_+ \).
Lagrangian Relaxation: Lagrangian

the dual can be computed explicitly as an (unconstrained) quadratic minimization problem, with:

\[
\inf_{x \in \mathbb{R}} x^T P x + q^T x + r = \begin{cases} 
  r - \frac{1}{4} q^T P^+ q, & \text{if } P \succeq 0 \text{ and } q \in \mathcal{R}(P) \\
  -\infty, & \text{otherwise}
\end{cases}
\]

we have:

\[
\inf_x L(x, \lambda) = -\frac{1}{4} (q_0 + \sum_{i=1}^m \lambda_i q_i)^T (P_0 + \sum_{i=1}^m \lambda_i P_i)^+ (q_0 + \sum_{i=1}^m \lambda_i q_i) \\
+ \sum_{i=1}^m \lambda_i r_i + r_0
\]

where we recognize a Schur complement...
the dual of the QCQP is then given by:

\[
\begin{align*}
\text{maximize} & \quad \gamma + \sum_{i=1}^{m} \lambda_i r_i + r_0 \\
\text{subject to} & \quad \begin{bmatrix} (P_0 + \sum_{i=1}^{m} \lambda_i P_i) & (q_0 + \sum_{i=1}^{m} \lambda_i q_i)/2 \\ (q_0 + \sum_{i=1}^{m} \lambda_i q_i)^T/2 & -\gamma \end{bmatrix} \succeq 0 \\
\lambda_i & \geq 0, \quad i = 1, \ldots, m
\end{align*}
\]

which is a semidefinite program in the variable \( \lambda \in \mathbb{R}^m \) and can be solved efficiently

let us look at what happens when we use semidefinite duality to compute the dual of this last program...
Taking the dual again, we get an SDP is given by:

\[
\begin{align*}
\text{minimize} & \quad \text{Tr}(XP_0) + q_0^T x + r_0 \\
\text{subject to} & \quad \text{Tr}(XP_i) + q_i^T x + r_i \leq 0, \quad i = 1, \ldots, m \\
& \quad \begin{bmatrix} X & x^T \\ x & 1 \end{bmatrix} \succeq 0
\end{align*}
\]

in the variables \(X \in \mathbb{S}^n\) and \(x \in \mathbb{R}^n\)

- this is a convexification of the original program
- we have recovered the semidefinite relaxation in an “automatic” way
Lagrangian Relaxation: Boolean LS

using the previous technique, we can relax the original Boolean LS problem:

\[
\text{minimize} \quad \|Ax - b\|^2 \\
\text{subject to} \quad x_i^2 = 1, \quad i = 1, \ldots, n
\]

and relax it as an SDP:

\[
\text{minimize} \quad \text{Tr}(AX) + 2b^T Ax + b^T b \\
\text{subject to} \quad \begin{bmatrix} X & x^T \\ x & 1 \end{bmatrix} \succeq 0 \\
X_{ii} = 1, \quad i = 1, \ldots, n
\]

this program then produces a lower bound on the optimal value of the original Boolean LS program
Lagrangian Relaxation: Partitioning

the partitioning problem defined above is:

\[
\begin{align*}
\text{minimize} & \quad x^T W x \\
\text{subject to} & \quad x_i^2 = 1, \quad i = 1, \ldots, n
\end{align*}
\]

the variable \( x \) disappears from the relaxation, which becomes:

\[
\begin{align*}
\text{minimize} & \quad \text{Tr}(W X) \\
\text{subject to} & \quad X \succeq 0 \\
& \quad X_{ii} = 1, \quad i = 1, \ldots, n
\end{align*}
\]
Feasible points?

- Lagrangian relaxations only provide lower bounds on the optimal value
- how can we compute good feasible points?
- can we measure how suboptimal this lower bound is?
Randomization

the original QCQP:

\[
\begin{align*}
\text{minimize} & \quad x^T P_0 x + q_0^T x + r_0 \\
\text{subject to} & \quad x^T P_i x + q_i^T x + r_i \leq 0, \quad i = 1, \ldots, m
\end{align*}
\]

was relaxed into:

\[
\begin{align*}
\text{minimize} & \quad \text{Tr}(X P_0) + q_0^T x + r_0 \\
\text{subject to} & \quad \text{Tr}(X P_i) + q_i^T x + r_i \leq 0, \quad i = 1, \ldots, m \\
& \quad \begin{bmatrix} X & x^T \\ x & 1 \end{bmatrix} \succeq 0
\end{align*}
\]

- the last (Schur complement) constraint is equivalent to \( X - xx^T \succeq 0 \)

- hence, if \( x \) and \( X \) are the solution to the relaxed program, then \( X - xx^T \) is a covariance matrix...
Randomization

• pick $x$ as a Gaussian variable with $x \sim \mathcal{N}(x, X - xx^T)$

• $x$ will solve the QCQP ”on average” over this distribution

in other words:

$$\begin{aligned}
\text{minimize} & \quad \mathbb{E}[x^T P_0 x + q_0^T x + r_0] \\
\text{subject to} & \quad \mathbb{E}[x^T P_i x + q_i^T x + r_i] \leq 0, \quad i = 1, \ldots, m
\end{aligned}$$

a good feasible point can then be obtained by sampling enough $x$ . . .
Linearization

Consider the constraint:

\[ x^T P x + q^T x + r \leq 0 \]

we decompose the matrix \( P \) into its positive and negative parts

\[ P = P_+ - P_- , \quad P_+, P_- \succeq 0 \]

and original constraint becomes

\[ x^T P_+ x + q_0^T x + r_0 \leq x^T P_- x \]
Linearization

both sides of the inequality are now convex quadratic functions. We linearize the right hand side around an initial feasible point $x_0$ to obtain

$$x^T P_+ x + q_0^T x + r_0 \leq x^{(0)}^T P_- x^{(0)} + 2x^{(0)}^T P_-(x - x^{(0)})$$

- the right hand side is now an affine lower bound on the original function $x^T P_- x$ (see §3.1.3 in the 364 reader).

- the resulting constraint is convex and more conservative than the original one, hence the feasible set of the new problem will be a convex subset of the original feasible set

- we form a convex restriction of the problem

We can then solve the convex restriction to get a better feasible point $x^{(1)}$ and iterate...
Bounds on suboptimality

• in certain particular cases, it is possible to get a hard bound on the gap between the optimal value and the relaxation result

• a classic example is that of the MAXCUT bound

The MAXCUT problem is a particular case of the partitioning problem:

\[
\begin{align*}
\text{maximize} & \quad x^T W x \\
\text{subject to} & \quad x_i^2 = 1, \quad i = 1, \ldots, n
\end{align*}
\]

its Lagrangian relaxation is computed as:

\[
\begin{align*}
\text{maximize} & \quad \text{Tr}(WX) \\
\text{subject to} & \quad X \succeq 0 \\
& \quad X_{ii} = 1, \quad i = 1, \ldots, n
\end{align*}
\]
Bounds on suboptimality: MAXCUT

Let \( X \) be a solution to this program

- we look for a feasible point by sampling a normal distribution \( \mathcal{N}(0, X) \)
- we convert each sample point \( x \) to a feasible point by rounding it to the nearest value in \( \{-1, 1\} \), i.e. taking
  \[
  \hat{x} = \text{sgn}(x)
  \]

crucially, when \( \hat{x} \) is sampled using that procedure, the expected value of the objective \( \mathbb{E}[\hat{x}^T W x] \) can be computed explicitly:

\[
\mathbb{E}[\hat{x}^T W x] = \frac{2}{\pi} \sum_{i,j=1}^{n} W_{ij} \arcsin(X_{ij}) = \frac{2}{\pi} \text{Tr}(W \arcsin(X))
\]
Bounds on suboptimality: MAXCUT

- we are guaranteed to reach this expected value $\frac{2}{\pi} \text{Tr}(W \arcsin(X))$ after sampling a few (feasible) points $\hat{x}$

- hence we know that the optimal value $OPT$ of the MAXCUT problem is between $\frac{2}{\pi} \text{Tr}(W \arcsin(X))$ and $\text{Tr}(WX)$

Furthermore, with $\arcsin(X) \succeq X$, we can simplify (and relax) the above expression to get:

$$\frac{2}{\pi} \text{Tr}(WX) \leq OPT \leq \text{Tr}(WX)$$

The procedure detailed above guarantees that we can find a feasible point at most $\frac{2}{\pi}$ suboptimal
Numerical Example: Boolean LS

Boolean least-squares problem:

\[
\begin{align*}
\text{minimize} & \quad \|Ax - b\|^2 \\
\text{subject to} & \quad x_i^2 = 1, \quad i = 1, \ldots, n
\end{align*}
\]

with

\[
\|Ax - b\|^2 = x^T A^T A x - 2b^T A x + b^T b = \text{Tr} A^T A X - 2b^T A^T x + b^T b
\]

where \( X = xx^T \), hence can express BLS as

\[
\begin{align*}
\text{minimize} & \quad \text{Tr} A^T A X - 2b^T A x + b^T b \\
\text{subject to} & \quad X_{ii} = 1, \quad X \succeq xx^T, \quad \text{rank}(X) = 1
\end{align*}
\]

\ldots still a very hard problem
SDP relaxation for BLS

using Lagrangian relaxation, remember:

\[ X \succeq xx^T \iff \begin{bmatrix} X & x \\ x^T & 1 \end{bmatrix} \succeq 0 \]

we obtained the SDP relaxation (with variables \( X, x \))

\[
\begin{align*}
\text{minimize} & \quad \text{Tr } A^TAX - 2b^TA^Tx + b^Tb \\
\text{subject to} & \quad X_{ii} = 1, \quad \begin{bmatrix} X & x \\ x^T & 1 \end{bmatrix} \succeq 0
\end{align*}
\]

• optimal value of SDP gives lower bound for BLS
• if optimal matrix is rank one, we’re done
Interpretation via randomization

- can think of variables $X, x$ in SDP relaxation as defining a normal distribution $z \sim \mathcal{N}(x, X - xx^T)$, with $\mathbb{E} z_i^2 = 1$
- SDP objective is $\mathbb{E} \|Az - b\|^2$

suggests randomized method for BLS:

- find $X^{opt}, x^{opt}$, optimal for SDP relaxation
- generate $z$ from $\mathcal{N}(x^{opt}, X^{opt} - x^{opt}x^{opt}^T)$
- take $x = \text{sgn}(z)$ as approximate solution of BLS (can repeat many times and take best one)
Example

- (randomly chosen) parameters $A \in \mathbb{R}^{150 \times 100}$, $b \in \mathbb{R}^{150}$
- $x \in \mathbb{R}^{100}$, so feasible set has $2^{100} \approx 10^{30}$ points

**LS approximate solution:** minimize $\|Ax - b\|$ s.t. $\|x\|^2 = n$, then round yields objective 8.7% over SDP relaxation bound

**randomized method:** (using SDP optimal distribution)

- best of 20 samples: 3.1% over SDP bound
- best of 1000 samples: 2.6% over SDP bound
\[ \frac{\|Ax - b\|}{\text{(SDP bound)}} \]

- SDP bound
- LS solution
we go back now to the two-way partitioning problem considered in exercise 5.39 of the reader:

\[
\begin{align*}
\text{minimize} & \quad x^T W x \\
\text{subject to} & \quad x_i^2 = 1, \quad i = 1, \ldots, n
\end{align*}
\]

the Lagrange dual of this problem is given by the SDP:

\[
\begin{align*}
\text{maximize} & \quad -1^T \nu \\
\text{subject to} & \quad W + \text{diag}(\nu) \succeq 0
\end{align*}
\]
Partitioning: Lagrangian relaxation

the dual of this SDP is the new SDP:

\[
\begin{align*}
\text{minimize} & \quad \text{Tr} \ W \ X \\
\text{subject to} & \quad X \succeq 0 \\
& \quad X_{ii} = 1, \quad i = 1, \ldots, n
\end{align*}
\]

the solution $X^{\text{opt}}$ gives a lower bound on the optimal value $p^{\text{opt}}$ of the partitioning problem.
Partitioning: simple heuristic

- solve the previous SDP to find $X^{opt}$ (and the bound $p^{opt}$)

- let $v$ denote an eigenvector of $X^{opt}$ associated with its largest eigenvalue

- now let

$$\hat{x} = \text{sgn}(v)$$

the vector $\hat{x}$ is our guess for a good partition
Partitioning: Randomization

- we generate independent samples $x^{(1)}, \ldots, x^{(K)}$ from a normal distribution with zero mean and covariance $X^{opt}$

- for each sample we consider the heuristic approximate solution

$$\hat{x}^{(k)} = \text{sgn}(x^{(k)})$$

- we then take the one with lowest cost
Partitioning: Numerical Example

we compare the performance of these methods on a randomly chosen problem

- the optimal SDP lower bound $p^\text{opt}$ is equal to $-1641$
- the simple $\text{sign}(x)$ heuristic gives a partition with total cost $-1280$

exactly what the optimal value is, we can’t say; all we can say at this point is that it is between $-1641$ and $-1280$
histogram of the objective obtained by the randomized heuristic, over 1000 samples: the minimum value reached here is $-1328$
we’re not sure what the optimal cost is, but now we know it’s between \(-1641\) and \(-1328\)
Greedy method

we can improve these results a little bit using the following simple greedy heuristic

- suppose the matrix $Y = \hat{x}^T W \hat{x}$ has a column $j$ whose sum $\sum_{i=1}^{n} y_{ij}$ is positive

- switching $\hat{x}_j$ to $-\hat{x}_j$ will decrease the objective by $2 \sum_{i=1}^{n} y_{ij}$

- if we pick the column $y_{j0}$ with largest sum, switch $\hat{x}_{j0}$ and repeat until all column sums are negative, we decrease the objective

if we apply this simple method to the solution of the randomized method we get an objective value of $-1356$, while applying it to the SDP heuristic gives an objective value of $-1372$, our best partition yet...