Semidefinite programming (SDP) has been developed to solve a wide range of problems in Engineering and Optimization. Although it possesses beautiful theoretical features and properties, the speed of SDP algorithms is unsatisfactory and un-scalable for many practical applications. In this project we explore methods to develop more efficient SDP models; more precisely, to decompose the single semidefinite matrix cone into a set of small-size semidefinite matrix cones, which we call the “smaller” SDP (SSDP) cone approach.

First some math notations. $\mathbb{R}^d$ denotes the $d$-dimensional Euclidean space, $S^n$ denotes the space of $n \times n$ symmetric matrices, $^T$ denotes transpose and $r(A)$ denotes the rank of $A$. For $A \in S^n$, $A_{ij}$ denotes the $(i,j)$th entry of $A$, and $A_{(i_1, \ldots, i_k), (i_1, \ldots, i_k)}$ denotes the principal submatrix of $A$ with the rows and columns indexed by $i_1, \ldots, i_k$. And for $A, B \in S^n$, $A \succeq B$ means that $A - B$ is positive semi-definite.

Given a general SDP problem:

$$\begin{align*}
\min & \quad C \cdot X \\
\text{s.t.} & \quad A_j \cdot X = b_j, \quad \forall j \\
& \quad X \succeq 0,
\end{align*}$$

where $X \in S^n$. We now consider a "decomposed" version of the problem:

$$\begin{align*}
\min & \quad C \cdot X \\
\text{s.t.} & \quad A_j \cdot X = b_j, \quad \forall j \\
& \quad X_{N_i, N_i} \succeq 0, \quad \forall i,
\end{align*}$$

where $N_i$ is an index subset of $\{1, 2, \ldots, n\}$, and could be chosen from the problem data structure and sparsity pattern. Note that here $N_i$s may not be disjoined.

**Question 1**: What is the dual of Problem (1)? What are the complementarity conditions?

Given a general SDP problem in the dual format:

$$\begin{align*}
\max & \quad \sum_j b_j y_j \\
\text{s.t.} & \quad \sum_j y_j A_j + S = C \\
& \quad S \succeq 0.
\end{align*}$$

We again consider a "decomposed" version of the problem:

$$\begin{align*}
\max & \quad \sum_j b_j y_j \\
\text{s.t.} & \quad \sum_j y_j A_j + S = C \\
& \quad S_{N_i, N_i} \succeq 0, \quad \forall i.
\end{align*}$$
Again $N_i$s may not be disjoined.

Note that the problem can be rewritten as

$$\text{max} \quad \sum_j b_j y_j$$

s.t.

$$\sum_j y_j (A_j)_{N_i,N_i} + S_{N_i,N_i} = C_{N_i,N_i},$$

$$S_{N_i,N_i} \succeq 0, \forall i. \quad (3)$$

This is a standard SDP problem solved by DSDP5.8.

**Question 2**: What is the dual of Problem (2) or (3)? What are the complementarity conditions? How to construct a feasible "solution" to the original SDP?

**Question 3**: When the decomposed version is equivalent to the original problem?

Some definitions may be useful.

**Definition 1.** A undirected graph is a chordal graph if every cycle of length greater than three has a chord; see, e.g., [2].

Also, the concept of partial positive semi-definite matrix can be found, e.g., in [5, 6, 7].

**Definition 2.** A square matrix is called to be partial symmetric if it is symmetric to the extent of its specified entries, i.e., if the $(i,j)$ entry of the matrix is specified, then so is the $(j,i)$ entry and the two are equal. A partial semi-definite matrix is a partial symmetric matrix and every fully specified principal submatrix is positive semi-definite.

The following results was proved in [6]

**Lemma 1.** Every partial positive semi-definite matrix with undirected graph $G$ has positive semi-definite completion if and only if $G$ is chordal.

**Question 4**: How good is the decomposed versions? How do they perform? Apply the decomposition method to solving either Max-Cut problems or the sensor localization problem, using Sedumi [10] or DSDP [1]. Explain your computational results and try to develop theorems to support the findings.

**Question 5**: Since these SDP decompositions have a special form, can you develop tailored interior-point methods to solve them?

**References**


