On Semidefinite Lift-and-Project relaxations for Combinatorial Optimization Problems

F. Battista[1]

Supervisors: M. De Santis[1], F. Rossi[2], S. Smriglio[2]

[1]Dipartimento di Ingegneria informatica, Automatica e Gestionale
Università di Roma "Sapienza"

[2]Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica
Università degli studi dell'Aquila

30th of November, 2022

The Stable Set Problem

Let $G = (V, E)$ be a simple graph with $V = \{1, \dots, n\}$ and E being its vertex and edge set (resp.)

A subset $S \subseteq V$ is **stable** iff all nodes in S are mutually not adjacent in G.

The **Stable Set Problem** (**SSP**) can be formulated as 0-1 LP:

$$\alpha(G) := \max \sum_{i \in V} x_i \quad \text{s.t.} \quad x_i + x_j \leq 1 \quad \forall \ \{i, j\} \in E, \quad x \in \{0, 1\}^n \tag{1}$$

The **SSP** is strongly NP-Hard [Håstad 1999] (equivalent to **Max Clique**).

The Stable Set Problem

Let $G = (V, E)$ be a simple graph with $V = \{1, \dots, n\}$ and E being its vertex and edge set (resp.)

A subset $S \subseteq V$ is **stable** iff all nodes in S are mutually not adjacent in G.

The **Stable Set Problem** (**SSP**) can be formulated as 0-1 LP:

$$\begin{aligned} \alpha(G) := \quad \max \quad & \textstyle\sum_{i \in V} x_i \\ \text{s.t.} \quad & x_i + x_j \leq 1 \quad \forall \ \{i, j\} \in E \\ & x \in \{0, 1\}^n \end{aligned} \tag{1}$$

The **SSP** is strongly NP-Hard [Håstad 1999] (equivalent to **Max Clique**).

One can strengthen the LP relaxation of (1) FRAC(G) by including valid linear inequalities (e.g. *cliques*, *odd holes* [Padberg 1973; Trotter Jr 1975]).

On the Lovász Theta function

In [Lovász 1979], the **Theta function** of a graph $\theta(G)$ was introduced.

Let us consider a feasible vector $x \in \{0,1\}^n$ for (1). Then, we define the *augmented matrix*

$$Y = \begin{pmatrix} 1 \\ x \end{pmatrix} \begin{pmatrix} 1 \\ x \end{pmatrix}^\top = \begin{pmatrix} 1 & x^\top \\ x & xx^\top \end{pmatrix}. \qquad (2)$$

By definition,

- $\text{rank}(Y) = 1$;
- $Y \in \mathcal{S}^+_{n+1}$ (or $Y \succeq 0$);
- $x_i x_j = 0$ for $\{i,j\} \in E$;
- since $x_i^2 = x_i$, the first column equals the diagonal.

On the Lovász Theta function

Dropping the rank-1 constraint leads to the following Semidefinite (SDP) relaxation of the **SSP**

$$\theta(G) = \max \quad \sum_{i\in V} x_i$$
$$\text{s.t.} \quad X_{ii} = x_i \quad \forall\, i \in V$$
$$X_{ij} = 0 \quad \forall\, \{i,j\} \in E \qquad \textbf{(th-SDP)}$$
$$\begin{pmatrix} 1 & x^\top \\ x & X \end{pmatrix} \succeq 0,$$

with $X \in \mathcal{S}_n$ (i.e. symmetric).

SDPs can be solved in polynomial time to arbitrary fixed precision [Grötschel *et al.* 1981].

Moreover, Lovász proved that

$$\alpha(G) \leq \theta(G).$$

Improving the Theta function: Previous works

To strengthen $\theta(G)$, one can add valid linear inequalities to (**th-SDP**).

- The inclusion of inequalities

$$X_{ij} \geq 0 \quad \forall \ \{i, j\} \notin E, \tag{3}$$

yields the formulation **th-SDP**$_+$ whit upper bound $\theta^+(G)$ [Schrijver 1979].

$$\alpha(G) \leq \qquad \theta^+(G) \leq \theta(G)$$

Improving the Theta function: Previous works

To strengthen $\theta(G)$, one can add valid linear inequalities to (**th-SDP**).

- The inclusion of inequalities

$$X_{ij} \geq 0 \quad \forall \ \{i,j\} \notin E, \tag{3}$$

yields the formulation **th-SDP**$_+$ whit upper bound $\theta^+(G)$ [Schrijver 1979].

- Further inclusion of

$$\begin{aligned} X_{ik} + X_{jk} &\leq x_k, \\ x_i + x_j + x_k &\leq 1 + X_{ik} + X_{jk}, \end{aligned} \quad \forall \ \{i,j\} \in E, k \neq i,j \tag{4}$$

yields the upper bound LS(G) [Lovász and Schrijver 1991].

$$\alpha(G) \leq \text{LS}(G) \leq \theta^+(G) \leq \theta(G)$$

Remark

(4) arise from the application of Lovász-Schrijver $M_+(\cdot)$ operator to FRAC(G).

Lift-and-Project: Lovász and Schrijver's $M_+(\cdot)$ and $N_+(\cdot)$

[Lovász and Schrijver 1991] Let K be the convex hull of integer solutions of some 0-1 LP, along with its relaxation

$$L := \{x \in [0,1]^n \ : \ Ax \leq b\} \supseteq K.$$

Lift-and-Project: Lovász and Schrijver's $M_+(\cdot)$ and $N_+(\cdot)$

[Lovász and Schrijver 1991] Let K be the convex hull of integer solutions of some 0-1 LP, along with its relaxation

$$L := \{x \in [0,1]^n \ : \ Ax \leq b\} \supseteq K.$$

1. For $i = 1, \ldots, n$ generate the set of non-linear inequalities

$$x_i(Ax - b) \leq 0$$
$$(1 - x_i)(Ax - b) \leq 0$$

2. Linearize using the augmented matrix Y and the substitutions:

$$Y = \begin{pmatrix} 1 & x^\top \\ x & X \end{pmatrix} \qquad x_i x_j = X_{ij}$$
$$x_i^2 = x_i$$

   The resulting feasible region is denoted by $M_+(L)$.

3. The projection of $M_+(L)$ onto the x-space is denoted by $N_+(L)$, with

$$K \subseteq N_+(L) \subseteq L.$$

Improving the Theta function (cont.)

Experiments [Burer *et al.* 2006, Dukanovic *et al.* 2007] draw the following picture:

- $\theta(G)$ is often significantly stronger than LP relaxation.
- A substantial improvement over $\theta(G)$ is usually paid with a considerable additional computational cost
- The inclusion of inequalities (4) to (**th-SDP**) produces SDPs hard to solve with general-purpose methods and they often require specialized algorithms
- As a consequence, stronger bound $\mathrm{LS}(G)$ is often computationally inaccessible on large instances.

Improving the Theta function (cont.)

Experiments [Burer *et al.* 2006, Dukanovic *et al.* 2007] draw the following picture:

- $\theta(G)$ is often significantly stronger than LP relaxation.
- A substantial improvement over $\theta(G)$ is usually paid with a considerable additional computational cost
- The inclusion of inequalities (4) to (**th-SDP**) produces SDPs hard to solve with general-purpose methods and they often require specialized algorithms
- As a consequence, stronger bound $\mathrm{LS}(G)$ is often computationally inaccessible on large instances.

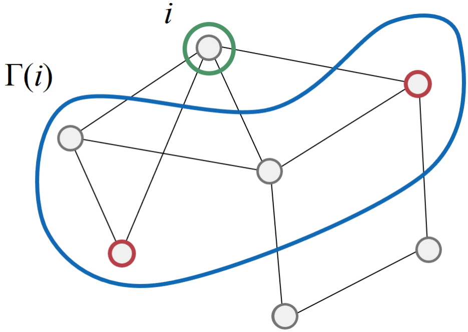
Question

Can we do better using the Lovász-Schrijver Lift-and-Project operator?

An alternative formulation for SSP

A hierarchy of LP formulations for the SSP can be obtained with the **Nodal inequalities** [Murray and Church 1997, Della Croce and Tadei 1994]

$$\sum_{j \in \Gamma(i)} x_j + r_i x_i \leq r_i \quad \forall i \in V, \tag{5}$$



An alternative formulation for SSP

A hierarchy of LP formulations for the SSP can be obtained with the **Nodal inequalities** [Murray and Church 1997, Della Croce and Tadei 1994]

$$\sum_{j \in \Gamma(i)} x_j + r_i x_i \leq r_i \quad \forall i \in V, \tag{5}$$

with $r_i \geq \alpha(G[\Gamma(i)])$. Possible values are:

$$\alpha(G[\Gamma(i)]) \quad \leq \quad \lfloor \theta(G[\Gamma(i)]) \rfloor \quad \leq \quad |\Gamma(i)|.$$

We define the following polytope:

$$\mathrm{NOD}_\alpha(G) := \left\{ x \in [0,1]^n \ : \ \sum_{j \in \Gamma(i)} x_j + \alpha(G[\Gamma(i)]) x_i \leq \alpha(G[\Gamma(i)]) \quad \forall i \in V \right\}.$$

Accordingly, we define $\mathrm{NOD}_\theta(G)$ and $\mathrm{NOD}_\Gamma(G)$.

Lifting the Nodal polytope: $M_+(\text{NOD}(G))$

Questions

1. Can the operator $M_+(\cdot)$ improve the bound of $\text{NOD}_{\Gamma|\theta|\alpha}(G)$?
2. What are the consequences of different choice of r_i?

Example: Let us choose some $i \in V$ and its variable x_i:

$$\sum_{j\in\Gamma(i)} x_j + r_i x_i - r_i \quad \leq 0$$

Lifting the Nodal polytope: $M_+(\text{NOD}(G))$

Questions

1. Can the operator $M_+(\cdot)$ improve the bound of $\text{NOD}_{\Gamma|\theta|\alpha}(G)$?
2. What are the consequences of different choice of r_i?

Example: Let us choose some $i \in V$ and its variable x_i:

$$x_i \left(\sum_{j \in \Gamma(i)} x_j + r_i x_i - r_i \right) \leq 0$$

Lifting the Nodal polytope: $M_+(\text{NOD}(G))$

Questions

1. Can the operator $M_+(\cdot)$ improve the bound of $\text{NOD}_{\Gamma|\theta|\alpha}(G)$?
2. What are the consequences of different choice of r_i?

Example: Let us choose some $i \in V$ and its variable x_i:

$$x_i \left(\sum_{j \in \Gamma(i)} x_j + r_i x_i - r_i \right) \leq 0$$

$$\sum_{j \in \Gamma(i)} x_i x_j + r_i x_i^2 - r_i x_i \leq 0.$$

Linearize by $x_i x_j = X_{ij},\ x_i^2 = x_i$.

Lifting the Nodal polytope: $M_+(\text{NOD}(G))$

Questions

1. Can the operator $M_+(\cdot)$ improve the bound of $\text{NOD}_{\Gamma|\theta|\alpha}(G)$?
2. What are the consequences of different choice of r_i?

Example: Let us choose some $i \in V$ and its variable x_i:

$$x_i \left(\sum_{j \in \Gamma(i)} x_j + r_i x_i - r_i \right) \leq 0$$

$$\sum_{j \in \Gamma(i)} x_i x_j + r_i x_i^2 - r_i x_i \leq 0.$$

Linearize by $x_i x_j = X_{ij}, \; x_i^2 = x_i$.

We obtain:

$$\sum_{j \in \Gamma(i)} X_{ij} \leq 0$$

A new hierarchy of SDP relaxations: $M_+(\text{NOD}_{\Gamma|\theta|\alpha}(G))$

Consider the following subset of constraints produced by $M_+(\cdot)$:

$$X_{ij} \geq 0 \qquad \forall i,j \in V, i \neq j \qquad (6)$$

$$\sum_{j\in\Gamma(i)} X_{ij} \leq 0 \qquad \forall\ i \in V \qquad (7)$$

$$\sum_{j\in\Gamma(i)\setminus\{k\}} X_{jk} \leq (r_i - 1)x_k \qquad \forall i,k \in V, k \in \Gamma(i) \qquad (8)$$

$$\sum_{j\in\Gamma(i)} X_{jk} \leq r_i x_k - r_i X_{ik} \qquad \forall i,k \in V, k \notin \Gamma(i) \qquad (9)$$

$$X_{ii} = x_i \qquad \forall\ i \in V \qquad (10)$$

$$Y = \begin{pmatrix} 1 & x^\top \\ x & X \end{pmatrix} \succeq 0$$

Remarks:

- (6) and (7) are independent from r_i, while (8) and (9) are not
- (10) are added from the operator to enforce the structure of Y

Main Result

Lemma

For any value $r_i \geq \alpha(G[\Gamma(i)])$, $\mathrm{OPT}(M_+(\mathrm{NOD}_r(G))) \leq \theta^+(G)$.

Proof: Let us consider the following relaxation of $M_+(\mathrm{NOD}_r(G))$:

$$
\begin{aligned}
\max \sum_{i \in V} x_i & & \\
\text{s.t. } X_{ij} \geq 0 & \quad \forall i, j \in V, i \neq j & (6)\\
\sum_{j \in \Gamma(i)} X_{ij} \leq 0 & \quad \forall i \in V & (7)\\
X_{ii} = x_i & \quad \forall\ i \in V & (10)\\
Y \succeq 0 & &
\end{aligned}
$$

Main Result

Lemma

For any value $r_i \geq \alpha(G[\Gamma(i)])$, $\mathrm{OPT}(M_+(\mathrm{NOD}_r(G))) \leq \theta^+(G)$.

Proof: Let us consider the following relaxation of $M_+(\mathrm{NOD}_r(G))$:

th-SDP$^+$

$$
\begin{aligned}
\max \; & \sum_{i \in V} x_i \\
\text{s.t. } & X_{ij} \geq 0 && \forall i, j \in V, i \neq j \\
& X_{ij} = 0 && \forall \; \{i,j\} \in E \\
& X_{ii} = x_i && \forall \; i \in V \\
& Y \succeq 0
\end{aligned}
$$

$$
\begin{aligned}
\max \; & \sum_{i \in V} x_i \\
\text{s.t. } & X_{ij} \geq 0 && \forall i, j \in V, i \neq j \quad (6) \\
& \sum_{j \in \Gamma(i)} X_{ij} \leq 0 && \forall i \in V \quad (7) \\
& X_{ii} = x_i && \forall \; i \in V \quad (10) \\
& Y \succeq 0
\end{aligned}
$$

Inequalities (6) and (7) imply $X_{ij} = 0 \quad \forall \; \{i,j\} \in E$.

Main Result

Lemma

For any value $r_i \geq \alpha(G[\Gamma(i)])$, $\text{OPT}(M_+(\text{NOD}_r(G))) \leq \theta^+(G)$.

Proof: Let us consider the following relaxation of $M_+(\text{NOD}_r(G))$:

th-SDP$^+$

$$
\begin{aligned}
\max \sum_{i \in V} x_i & \\
\text{s.t. } X_{ij} \geq 0 & \quad \forall i,j \in V, i \neq j \\
X_{ij} = 0 & \quad \forall \ \{i,j\} \in E \\
X_{ii} = x_i & \quad \forall \ i \in V \\
Y \succeq 0 &
\end{aligned}
$$

$$
\begin{aligned}
\max \sum_{i \in V} x_i & \\
\text{s.t. } X_{ij} \geq 0 & \quad \forall i,j \in V, i \neq j \ (6) \\
X_{ij} = 0 & \quad \forall \ \{i,j\} \in E \\
X_{ii} = x_i & \quad \forall \ i \in V \ (10) \\
Y \succeq 0 &
\end{aligned}
$$

Inequalities (6) and (7) imply $X_{ij} = 0 \quad \forall \ \{i,j\} \in E$.
This implies that $M_+(\text{NOD}_r(G))$ is at least as restrictive as **th-SDP$^+$**. □

Numerical results: Setup

Both $M_+(\text{FRAC}(G))$ and $M_+(\text{NOD}_{\alpha|\theta|\Gamma}(G))$ are strengthenings of **th-SDP**$^+$.

Algorithm Kelley's cutting-plane scheme [Kelley 1960]

1: Let $\Pi = \emptyset$
2: Solve **th-SDP**$^+$
3: **repeat**
4: Include to Π the 1000 most violated cuts from the current solution
5: Solve **th-SDP**$^+ \cap \Pi$
6: **until** No more *violated cuts* are identified **or**
7: The *objective value* is not improving substantially

SDP Solver:

- ADMM SDPNAL+ [Yang, Sun, and Toh 2015] (MATLAB)

Instances:

- Erdös–Rényi $G(n, p)$ random graphs from [Letchford *et al.* 2020],
- DIMACS Second Challenge instances [Johnson and Trick 1996]

Numerical results: Random Instances (1)

		$\theta^+(G)$	$M_+(\mathrm{NOD}_\Gamma(G))$			$M_+(\mathrm{NOD}_\theta(G))$			$M_+(\mathrm{NOD}_\alpha(G))$		
n	p	Gap	Gap	Cuts	CPU-time	Gap	Cuts	CPU-time	Gap	Cuts	CPU-time
150	1	12.954	12.953	1.4	7.822	12.952	6.4	10.662	12.952	6.4	9.264
	2	20.269	20.269	0.0	2.894	20.269	0.2	3.972	20.269	0.2	3.420
	3	22.330	22.330	0.0	1.962	22.330	1.0	2.520	22.330	1.0	2.378
	4	28.268	28.268	0.0	1.282	28.268	2.4	2.604	**28.266**	6.8	2.632
	5	24.861	24.861	0.0	1.186	**24.809**	34.6	3.146	**24.362**	193.2	3.514
	6	21.161	21.161	0.0	1.300	**19.859**	539.2	8.144	**17.281**	1095.4	10.900
	7	14.027	14.027	0.0	1.334	**6.694**	1451.4	16.066	**6.200**	1479.0	15.202
	8	10.140	10.140	0.0	2.652	**3.441**	1250.2	15.570	**3.441**	1250.2	14.856
	9	1.378	1.378	0.0	13.492	**1.071**	926.0	49.826	**1.071**	926.0	54.250
225	1	22.934	22.934	0.0	7.800	22.934	0.0	7.896	22.934	0.0	7.564
	2	32.506	32.506	0.0	5.034	32.506	0.0	4.636	32.506	0.0	4.304
	3	34.482	34.482	0.0	3.274	34.482	0.0	3.320	34.482	0.2	3.622
	4	38.547	38.547	0.0	2.170	38.547	0.4	2.530	**38.524**	20.2	4.282
	5	35.876	35.876	0.0	1.828	**35.873**	4.8	3.524	**33.947**	1088.8	14.466
	6	32.497	32.497	0.0	1.918	**32.133**	255.2	5.876	**24.985**	2000.0	18.024
	7	26.023	26.023	0.0	1.790	**20.533**	1903.2	21.770	**13.745**	2825.2	34.120
	8	15.208	15.208	0.0	2.746	**8.088**	2000.0	24.756	**0.149**	1914.2	194.072
	9	12.293	12.293	0.0	5.328	**4.074**	2600.0	33.162	**4.074**	2600.0	32.476
300	1	31.124	31.124	0.0	10.250	31.124	0.0	10.006	31.124	0.0	10.370
	2	46.238	46.238	0.0	7.538	46.238	0.0	7.160	46.238	0.0	7.416
	3	48.426	48.426	0.0	5.428	48.426	0.0	5.534	48.426	0.0	5.330
	4	50.228	50.228	0.0	3.266	50.228	0.0	3.254	**50.166**	76.8	6.572
	5	46.061	46.061	0.0	2.334	46.061	0.0	2.652	**41.439**	2000.0	23.406
	6	39.956	39.956	0.0	2.412	**39.850**	129.4	6.038	**26.444**	3000.0	42.344
	7	31.619	31.619	0.0	2.284	**28.331**	1813.6	25.180	**11.347**	4000.0	67.134
	8	29.809	29.809	0.0	3.614	**19.988**	2200.0	41.766	**11.667**	3600.0	69.064
	9	23.160	23.160	0.0	6.256	**6.224**	3800.0	90.712	**5.681**	4200.0	97.612

Numerical results: Random Instances (1)

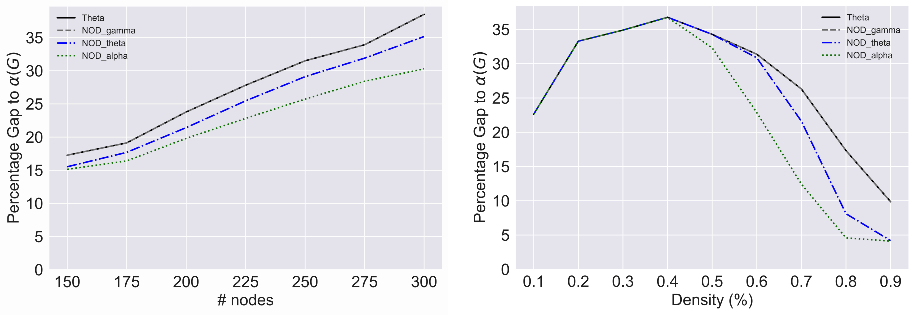
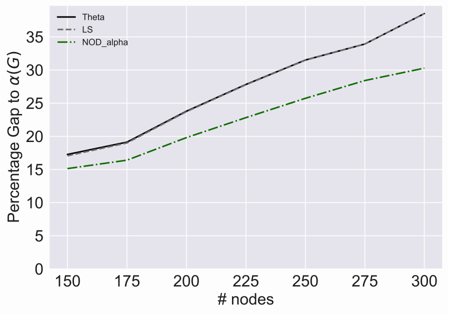


Figure: Average percentage gap of SDP bounds on Erdös–Rényi $G(n,p)$ random graphs

Numerical results: Random Instances (2)

		$\text{NOD}_\alpha(G)$	$\theta^+(G)$		$M_+(\text{NOD}_\alpha(G))$			$M_+(\text{FRAC}(G))$		
n	p	Gap	Gap	CPU-time	Gap	Cuts	CPU-time	Gap	Cuts	CPU-time
150	0.1	32.117	12.954	4.358	12.952	6.4	9.264	**11.369**	1311.8	17.888
	0.2	75.793	20.269	2.910	20.269	0.2	3.420	**20.109**	187.2	5.642
	0.3	82.288	22.330	1.720	22.330	1.0	2.378	**22.316**	27.0	3.154
	0.4	75.183	28.268	1.144	28.266	6.8	2.632	28.265	5.6	2.456
	0.5	56.020	24.861	1.096	**24.362**	193.2	3.514	24.860	1.6	2.230
	0.6	39.308	21.161	1.332	**17.281**	1095.4	10.900	21.161	0.6	1.988
	0.7	23.890	14.027	1.280	**6.200**	1479.0	15.202	14.026	0.8	2.434
	0.8	14.920	10.140	2.954	**3.441**	1250.2	14.856	10.137	1.4	4.442
	0.9	8.900	1.378	15.110	**1.071**	926.0	54.250	1.341	8.0	27.070
225	0.1	54.825	22.934	7.556	22.934	0.0	7.564	**22.559**	798.4	27.196
	0.2	112.700	32.506	4.294	32.506	0.0	4.304	32.503	11.4	9.216
	0.3	97.343	34.482	3.214	34.482	0.2	3.622	34.482	0.6	4.530
	0.4	81.214	38.547	2.134	**38.524**	20.2	4.282	38.547	0.0	2.370
	0.5	59.427	35.876	1.876	**33.947**	1088.8	14.466	35.876	0.2	2.504
	0.6	43.044	32.497	1.860	**24.985**	2000.0	18.024	32.497	0.0	2.212
	0.7	29.957	26.023	1.782	**13.745**	2825.2	34.120	26.023	0.0	2.232
	0.8	13.733	15.208	2.730	**0.149**	1914.2	194.072	15.208	0.0	3.172
	0.9	8.130	12.293	5.274	**4.074**	2600.0	32.476	12.293	0.0	5.812
300	0.1	75.661	31.124	10.342	31.124	0.0	10.370	**31.047**	256.2	23.402
	0.2	137.919	46.238	7.386	46.238	0.0	7.416	46.237	1.4	12.340
	0.3	113.088	48.426	5.288	48.426	0.0	5.330	48.426	0.0	5.694
	0.4	89.040	50.228	3.086	**50.166**	76.8	6.572	50.228	0.0	3.682
	0.5	62.433	46.061	2.556	**41.439**	2000.0	23.406	46.061	0.0	3.362
	0.6	42.224	39.956	2.474	**26.444**	3000.0	42.344	39.956	0.0	3.414
	0.7	23.650	31.619	2.216	**11.347**	4000.0	67.134	31.619	0.0	3.390
	0.8	23.333	29.809	3.426	**11.667**	3600.0	69.064	29.809	0.0	4.432
	0.9	7.090	23.160	6.302	**5.681**	4200.0	97.612	23.160	0.0	7.702

Numerical results: Random Instances (2)



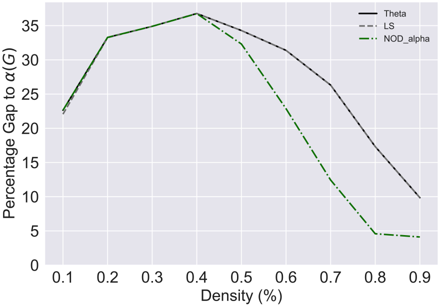


Figure: Average percentage gap of SDP bounds on Erdös–Rényi $G(n, p)$ random graphs

Numerical results: DIMACS Instances

	$\theta^+(G)$		$M_+(\text{NOD}_\alpha(G))$			$M_+(\text{FRAC}(G))$		
Graph	Gap	CPU-time	Gap	Cuts	CPU-time	Gap	Cuts	CPU-time
brock200_1	29.508	3.53	29.508	0	3.55	29.505	10	6.47
brock200_2	17.758	1.59	**16.795**	659	12.31	17.758	0	1.81
brock400_1	45.670	11.79	45.670	0	11.83	45.670	0	12.66
brock400_2	35.160	12.16	35.160	0	12.23	35.160	0	12.85
brock800_1	82.032	19.27	**80.646**	2000	64.28	-	-	-
brock800_2	75.435	19.33	**73.896**	2000	72.00	-	-	-
brock800_3	67.530	19.83	**66.043**	2000	63.54	-	-	-
brock800_4	61.541	19.17	**60.039**	2000	63.59	-	-	-
p_hat300-1	25.253	8.98	**7.288**	3000	120.48	25.253	0	9.95
p_hat300-2	6.855	80.75	**6.768**	727	310.26	**6.317**	988	206.70
p_hat500-1	44.533	17.27	**18.721**	5000	381.28	-	-	-
p_hat500-2	48.306	190.29	**48.225**	1000	918.25	**47.863**	1043	682.15
p_hat700-1	36.774	33.90	**2.709**	7000	2038.55	-	-	-
p_hat700-2	10.091	426.60	**10.023**	1000	1270.03	-	-	-
DSJC500-5	73.621	6.01	**58.014**	4000	85.91	73.621	0	9.79
hamming10-4	6.667	29.41	6.667	0	29.87	-	-	-
keller4	22.417	4.24	**22.236**	144	12.24	**22.388**	48	8.15
keller5	14.799	81.25	14.799	0	81.49	-	-	-
MANN_a27	5.367	9.26	**4.752**	1000	76.14	**4.057**	1212	631.01
sanr200_0.9	16.440	6.49	16.440	1	13.67	**15.786**	1060	27.41
sanr400_0.5	55.217	4.05	**46.952**	3000	52.79	55.217	0	5.89
sanr400_0.7	61.746	7.38	61.746	0	7.46	61.746	0	8.45

Conclusions

We introduced a new hierarchy of SDP relaxations for the Stable Set Problem:

- The first level is **at least as strong** as the Schrijver relaxation
- Stronger levels may **substantially** outperform $\theta^+(G)$ and LS(G) on dense graphs
- This behaviour scales well as the size of the graph increases
- `SDPNAL+` along with a Kelley's cutting plane allowed us to compute SDP bounds within a reasonable overhead of time w.r.t. $\theta^+(G)$

Conclusions

What's next?

Short-term question 1 (Ongoing)

Are there special graphs for which $N_+(\text{NOD}(G)) = \text{STAB}(G)$?
Related work: Definition of LS-perfect graphs in [Bianchi et al. 2017]

Short-term question 2 (Ongoing)

What about $M_+(\cdot)$ application to LP relaxations of **Graph Coloring Problem**?

Long-term question

Is there an algorithm to identify a subset of variables to which apply the lifting $M_+(\cdot)$ while still lead to a significant improvement of the bound?

SDPs with inequalities

We focus on SDPs in the following form:

$$\begin{aligned} \min \quad & \langle C, X\rangle \\ \text{s.t.} \quad & \langle A^i, X\rangle \le b_i, \qquad \forall\, i = 1,\dots,l \\ & \langle A^j, X\rangle = b_j, \qquad \forall\, j = l+1,\dots,m \\ & X \in \mathcal{S}_n^+ \end{aligned} \qquad \textbf{(SDP)}$$

where:

- $\langle M, N\rangle = \mathrm{tr}(MN)$ is the standard inner product in $\mathcal{S}_n$
- $C \in \mathcal{S}_n$,
- $A^i \in \mathcal{S}_n$, $i = 1,\dots,m$,
- $b \in \mathbb{R}^m$.

SDPs with inequalities

Reduce (**SDP**) in standard form and write the dual:

$$\begin{aligned} \min \quad & \langle \bar{C}, \bar{X} \rangle \\ \text{s.t.} \quad & \bar{\mathcal{A}} X = b \\ & \bar{X} \in \mathcal{S}^+_{n+l} \end{aligned}$$

where

- $\bar{\mathcal{A}} : \mathcal{S}_{n+l} \to \mathbb{R}^m$ with $(\bar{\mathcal{A}} X)_i = \langle \bar{A}^i, X \rangle$, $\bar{A}^i \in \mathcal{S}_{n+l}$, $i = 1, \dots, m$.

SDPs with inequalities

Reduce (**SDP**) in standard form and write the dual:

$$\begin{aligned} \min \quad & \langle \bar{C}, \bar{X} \rangle \\ \text{s.t.} \quad & \bar{\mathcal{A}} X = b \\ & \bar{X} \in \mathcal{S}^+_{n+l} \end{aligned} \qquad \begin{aligned} \max \quad & b^\top y \\ \text{s.t.} \quad & \bar{\mathcal{A}}^\top(y) + \bar{Z} = \bar{C} \\ & \bar{Z} \in \mathcal{S}^+_{n+l}, \end{aligned} \qquad \textbf{(SDPS)}$$

where

- $\bar{\mathcal{A}} : \mathcal{S}_{n+l} \to \mathbb{R}^m$ with $(\bar{\mathcal{A}} X)_i = \langle \bar{A}^i, X \rangle$, $\bar{A}^i \in \mathcal{S}_{n+l}$, $i = 1, \dots, m$.
- $\bar{\mathcal{A}}^\top : \mathbb{R}^m \to \mathcal{S}_{n+l}$ is the adjoint operator $\bar{\mathcal{A}}^\top(y) := \sum_i y_i \bar{A}_i$

On solving SDPs with ADMMs

Main tools for solving **SDPS**:

Interior-point methods [Nesterov and Nemirovskii 1994]

- Good precision and convergence for small/medium size SDPs
- Impractical for large scale SDPs due to memory requirements

Alternating Direction Methods of Multipliers [Malick et al. 2009]

- Better scalability on instances with large number of constraints
- May require more time to reach high accuracy

ADMMs are more suitable for solving SDPs obtained by $M_+(\cdot)$ operator.

ADAL: Alternating Direction Augmented Lagrangian

ADAL [Wen et al. 2010] optimizes the augmented Lagrangian of the dual **SDPS**:

$$\max_{y\in\mathbb{R}^m,\ \bar{Z}\in\mathcal{S}^+_{n+l}} L_\sigma(y,\bar{Z};\bar{X}) = b^T y - \langle \bar{\mathcal{A}}^\top(y) + \bar{Z} - \bar{C}, \bar{X}\rangle - \frac{\sigma}{2}\|\bar{\mathcal{A}}^\top(y) + \bar{Z} - \bar{C}\|^2.$$

At each iteration, the new point $(y^{k+1}, \bar{Z}^{k+1}, \bar{X}^{k+1})$ is given by:

$$y^{k+1} = \operatorname*{argmax}_{y\in\mathbb{R}^m} L_{\sigma^k}(y, \bar{Z}^k; \bar{X}^k), \tag{11}$$

$$\bar{Z}^{k+1} = \operatorname*{argmax}_{\bar{Z}\in\mathcal{S}^+_{n+l}} L_{\sigma^k}(y^{k+1}, \bar{Z}; \bar{X}^k), \tag{12}$$

$$\bar{X}^{k+1} = \bar{X}^k + \sigma^k(\mathcal{A}^\top(y^{k+1}) + \bar{Z}^{k+1} - \bar{C}). \tag{13}$$

ADAL: Alternating Direction Augmented Lagrangian

Algorithm Scheme of ADAL [Wen et al. 2010]

1: Choose $\sigma > 0$, $\bar{X}, \bar{Z} \in \mathcal{S}^+_{n+l}$
2: **repeat**
3: $\quad y = (\bar{\mathcal{A}}\bar{\mathcal{A}}^\top)^{-1}\left(\frac{1}{\sigma} b - \bar{\mathcal{A}}(\frac{1}{\sigma}\bar{X} - \bar{C} + \bar{Z})\right)$
4: $\quad \bar{W} := \bar{X}/\sigma - \bar{C} + \bar{\mathcal{A}}^\top(y)$
5: $\quad \bar{Z} = -(\bar{W})_-$
6: $\quad \bar{X} = \sigma(\bar{W})_+$
7: $\quad$ Update σ
8: **until** convergence

KKT conditions in ADAL:

- Satisfied at every iteration: $\bar{X} \succeq 0,\ \bar{Z} \succeq 0,\ \bar{X}\bar{Z} = 0;$
- Converges when: $\bar{\mathcal{A}}\bar{X} = b,\ \bar{\mathcal{A}}^\top(y) + \bar{Z} = \bar{C}.$

ADAL: Alternating Direction Augmented Lagrangian

Pre-existing implementation [Wen et al. 2010]:

- `MATLAB` (commercial)
- Solves SDPs in standard form
- No actual support for inequality constraints

Our implementation:

- `python` (open source)
- Solves SDPs with both inequalities and equalities

ADAL: Alternating Direction Augmented Lagrangian

Pre-existing implementation [Wen et al. 2010]:

- `MATLAB` (commercial)
- Solves SDPs in standard form
- No actual support for inequality constraints

Our implementation:

- `python` (open source)
- Solves SDPs with both inequalities and equalities

Experiment:

Comparison ADAL (`python`) vs `SDPNAL+` [Yang, Sun, and Toh 2015]

Instances:

Random SDPs from the generator proposed in [Malick et al. 2009] (total: 150 instances)

CPU time limit: 1800 secs

Numerical results: Random SDPs

			ADAL		SDPNAL+	
n	m	(% ineq.)	#sol	CPU-time	#sol	CPU-time
250	25000	25	5	**838.04**	0	-
		50	5	**1166.45**	0	-
		75	5	**1114.52**	0	-
500	50000	25	5	217.61	5	**106.28**
		50	5	260.43	5	**221.66**
		75	5	325.71	5	**250.97**
1000	10000	25	5	136.63	5	**49.52**
		50	5	157.21	5	**58.22**
		75	5	242.63	5	**71.38**
	50000	25	5	**57.19**	5	60.96
		50	5	**94.09**	5	109.48
		75	5	**110.00**	5	111.29
	100000	25	5	**83.15**	5	136.53
		50	5	**127.37**	5	181.13
		75	5	**155.05**	5	184.21

Table: Results on random instances

Safe bounding SDPs

When solving SDP relaxations of some Combinatorial Optimization problem, being able to compute safe bounds has a two-fold purpose:

- as a **post-processing** to "clean" the inaccuracy left by the ADMM;
- to stop prematurely the ADMM, when considering branch-and-bound frameworks, for example.

Let us consider the primal-dual pair:

$$\begin{array}{lll} \min & \langle C, X \rangle \\ \text{s.t.} & \mathcal{A}(X) = b \\ & X \succeq 0 \end{array} \qquad \begin{array}{lll} \max & b^\top y \\ \text{s.t.} & C - \mathcal{A}^\top(y) = Z \\ & Z \succeq 0 \end{array} \qquad \textbf{(SDP)}$$

Assumption: **strong duality** holds for **SDP**.

Dual safe bounds (DB)

By weak duality, any dual feasible solution (y, Z) provides a bound on the optimal primal value. Based on this, [Cerulli et al. 2021] proposed the following:

Let $\tilde{Z} \succeq 0$, then if the linear program:

$$\begin{aligned} \max \quad & b^\top y \\ \text{s.t.} \quad & C - \mathcal{A}^\top(y) = \tilde{Z} \\ & y \text{ free}, \end{aligned} \tag{D-LP}$$

is feasible, then $b^\top y^*$ is a safe bound on the primal **SDP**.

Dual safe bounds (DB)

By weak duality, any dual feasible solution (y, Z) provides a bound on the optimal primal value. Based on this, [Cerulli et al. 2021] proposed the following:

Let $\tilde{Z} \succeq 0$, then if the linear program:

$$\begin{aligned} \max \quad & b^\top y \\ \text{s.t.} \quad & C - \mathcal{A}^\top(y) = \tilde{Z} \\ & y \text{ free}, \end{aligned} \qquad \textbf{(D-LP)}$$

is feasible, then $b^\top y^*$ is a safe bound on the primal **SDP**.

Dual safe bound - Recap

What we need: $\tilde{Z} \succeq 0$
Computational burden: Solve **D-LP**
We are not guaranteed that the LP is always feasible.

Rigorous error bound (EB)

[Jansson et al. 2008] instead, proposed the following:

Let $\tilde{x} \in \mathbb{R}_+$ s.t. $\lambda_{\max}(X^*) \leq \tilde{x}$, with X^* an optimal primal sol. of **SDP**. Given any $y \in \mathbb{R}^m$, set

$$\tilde{Z} = C - \mathcal{A}^\top(y),$$

then it can be proved that:

$$\langle C, X^* \rangle \geq b^\top y + \sum_{i:\lambda_i(\tilde{Z})<0} \tilde{x}\lambda_i(\tilde{Z}).$$

Rigorous error bound (EB)

[Jansson et al. 2008] instead, proposed the following:

Let $\tilde{x} \in \mathbb{R}_+$ s.t. $\lambda_{\max}(X^*) \leq \tilde{x}$, with X^* an optimal primal sol. of **SDP**. Given any $y \in \mathbb{R}^m$, set

$$\tilde{Z} = C - \mathcal{A}^\top(y),$$

then it can be proved that:

$$\langle C, X^* \rangle \geq b^\top y + \sum_{i:\lambda_i(\tilde{Z})<0} \tilde{x}\lambda_i(\tilde{Z}).$$

Rigorous error bound - Recap

What we need: $\tilde{x}$ s.t. $\lambda_{\max}(X^*) \leq \tilde{x}$
Computational burden: Compute eigenvalues of $\tilde{Z}$
For structured SDPs, $\tilde{x}$ may be known a priori.

Norm bound (NB)

Joint work with [J. Schwiddessen 2023+]

Let X^* be an optimal solution of the primal. Suppose we know that $||X^*||_F \leq U$, for some U.

Hence, the optimal value of

$$\begin{aligned} \min \quad & \langle C, X \rangle \\ \text{s.t.} \quad & \mathcal{A}(X) = b \\ & ||X||_F \leq U \\ & X \succeq 0, \end{aligned} \qquad \textbf{(P-Norm)}$$

is the same of the primal **SDP**. By writing the Lagrangian of **P-Norm**, we can show that:

$$\text{OPT}(\textbf{P-Norm}) \geq b^\top y - U \left\| C - \mathcal{A}^\top(y) - Z \right\|_F,$$

for any $y \in \mathbb{R}^m, Z \succeq 0$.

Norm bound (NB)

Hence, the optimal value of

$$\begin{aligned} \min \quad & \langle C, X \rangle \\ \text{s.t.} \quad & \mathcal{A}(X) = b \\ & ||X||_F \leq U \\ & X \succeq 0, \end{aligned} \qquad \textbf{(P-Norm)}$$

is the same of the primal **SDP**. By writing the Lagrangian of **P-Norm**, we can show that:

$$\text{OPT}(\textbf{P-Norm}) \geq b^\top y - U \left\| C - \mathcal{A}^\top(y) - Z \right\|_F,$$

for any $y \in \mathbb{R}^m, Z \succeq 0$.

Norm bound - Recap

What we need: U s.t. $||X^*||_F \leq U$
Computational burden: Compute a norm.
Again, for structured SDPs U may be known a priori.

Numerical results

Question

Are the Safe bounding procedures able to identify a "good" bound within the time needed by ADAL to converge?

Setup:

- ADAL (`python`) + Safe bounding procedures (applied every 200 iterations)
- CPU-time limit: 3600 secs

Comparison:

- Best safe bounds found from **DB**, **EB** and **NB**
- CPU-time needed to identify the best bound
- Overhead

Instances:

- **th-SDP**$^+$ for Schrijver's number $\theta^+(G)$ on DIMACS instances

Remark: initial $\lambda_{\max}(X^*)$ and U can be obtained from $\frac{|V|}{2} \geq \alpha(G)$.

Numerical results

Graph	ADAL		Norm Bound		Error Bound			Dual Bound		
	ObjVal	CPU-time	best	found at	best	found at	overhead	best	found at	overhead
DSJC1000-5	31.67	33.85	31.93	**25.75**	55.76	26.19	0.43	**31.67**	32.62	4.56
C2000-5	44.56	150.67	45.31	**117.69**	136.03	119.01	1.32	**44.56**	145.41	18.95
C2000-9	177.73	2836.23	177.78	2672.59	**177.75**	2726.72	40.40	177.95	**1784.69**	194.49
brock800_1	41.87	33.97	41.88	**27.24**	88.37	27.49	0.50	**41.87**	35.24	2.20
brock800_2	42.10	34.87	42.11	**27.31**	90.01	27.56	0.49	**42.10**	36.18	2.23
brock800_3	41.88	33.46	41.89	**27.75**	84.12	27.99	0.48	**41.88**	34.74	2.19
brock800_4	42.00	34.81	42.01	**27.33**	86.63	27.57	0.50	**42.00**	36.11	2.25
p_hat1000-1	17.52	404.67	**17.52**	115.77	17.57	116.09	1.61	**17.52**	**88.78**	27.78
p_hat1000-2	54.84	2852.66	**54.84**	2851.17	54.85	2851.48	31.70	54.85	**302.03**	245.96
p_hat1000-3	83.53	2337.28	**83.53**	694.69	83.54	695.04	8.38	**83.53**	**242.76**	154.07
p_hat1500-1	21.89	1118.44	**21.89**	295.59	21.99	296.27	3.45	**21.89**	**216.50**	79.24
p_hat1500-2	-	-	**76.46**	3565.92	76.48	3494.53	34.54	**76.46**	**872.20**	234.78
p_hat1500-3	113.65	3014.54	113.66	3014.01	113.66	3014.70	35.88	**113.65**	**958.14**	226.53
keller5	31.00	1281.91	**31.00**	**499.35**	31.62	499.55	6.73	**31.00**	755.56	67.24
keller6	-	-	63.03	**1535.40**	288.88	1541.51	20.91	**63.00**	1913.78	79.56
MANN_a27	132.76	838.69	**132.76**	561.87	132.77	475.86	8.62	132.94	**340.53**	43.47
hamming6-2	32.00	11.54	32.75	**0.97**	**32.00**	5.95	0.03	**32.00**	6.54	0.50
hamming8-2	128.00	1951.74	128.53	**36.17**	**128.00**	1245.57	23.90	**128.00**	419.50	104.14
hamming10-4	42.67	60.55	**42.68**	50.41	42.76	**31.87**	2.99	42.76	33.11	4.28

Conclusions

We proposed a numerically stable implementation of ADAL:

- suited for SDPs with inequalities;
- Competitive with state-of-the-art ADMM;

Safe bounding procedures within an ADMM:

- overcome to inaccuracies left by the algorithm;
- allow to stop the execution prematurely;

What's next?

Short-term question (Ongoing)

Can we use ADAL + Safe bounding procedures within a Branch-and-Bound framework?

Long-term question

Can we find a good starting point for the ADMM to enhance the convergence? ("Reoptimization techniques")

Thanks for your attention!
Questions?