Applications of semidefinite programming, symmetry and algebra to graph partitioning problems

Edwin van Dam and Renata Sotirov

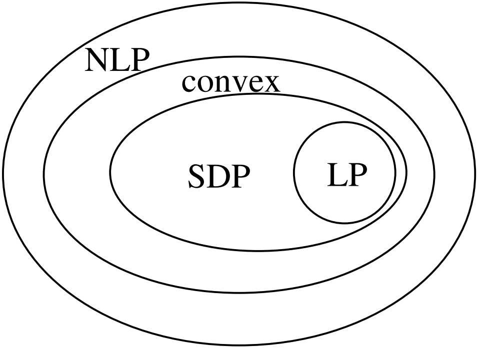
Tilburg University, The Netherlands

Novosibirsk, August 2018

Semidefinite programming ...

- generalization of linear programming (LP)
- unifies linear and quadratic programming problems
- arise naturally as relaxation of discrete optimization problems
- can be efficiently solved by interior-point-methods
- applications:
  - global and combinatorial optimization
  - eigenvalue optimization
  - robust optimization
  - circuit design
  - coding theory
  - finance
  - signal processing
  - chemical engineering
  - sensor network localization, etc.

Where is SDP?



Primal SDP

Primal problem:

$$\begin{aligned} \min \quad & \mathrm{tr}(CX) \\ \text{s.t.} \quad & \mathrm{tr}(A_i X) = b_i, \quad \forall\, i = 1, \ldots, m \\ & X \succeq 0 \end{aligned}$$

where $C, A_i \in \mathcal{S}_n$, $b_i \in \mathbb{R}$ $(i = 1, \ldots, m)$.

- $\mathcal{S}_n$ . . . space of symmetric $n \times n$ matrices
- $X \succeq 0$ . . . positive semidefinite iff $z^{\mathrm{T}} X z \geq 0$, $\forall z \in \mathbb{R}^n$

  iff all eigenvalues of X are ≥ 0

N.B. SDP reduces to LP when all matrices are diagonal.

Historical events related to SDP

- Lyapunov (1890)
  - stability of dynamic systems
- Bellman and Fan (1963)
  - first SDP formulated
- Lovász (1979)
  - upper bound Shannon capacity of a graph
- Lovász and Schrijver (1991)
  - SDP can provide tighter relaxations of 0-1 problems than LP

Historical events related to SDP

- Lyapunov (1890)
  - stability of dynamic systems
- Bellman and Fan (1963)
  - first SDP formulated
- Lovász (1979)
  - upper bound Shannon capacity of a graph
- Lovász and Schrijver (1991)
  - SDP can provide tighter relaxations of 0-1 problems than LP
- Goemans, Williamson (1995)
  - SDP-based approximation for max-cut

On solving SDP ...

POLYNOMIAL TIME ALGORITHMS:

- Ellipsoid method
  - Grötschel, Lovász and Schrijver (1988)
  - first to solve SDP in polynomial time
  - **not** practical
- Interior-point methods (IPM)
  - Nesterov and Nemirovski (1994), Alizadeh (1995)
  - practical, suitable for medium size
  - available software:
    - CSDP
    - DSDP
    - SDPA
    - SDPT3
    - SeDuMi
    - Mosek

$\Rrightarrow$ since 1995 the interest in SDP has grown tremendously

Max-cut

Given:

- $G = (V, E)$, ... an undirected graph with $|V| = n$
- $w_{ij} = w_{ji} \geq 0$ ... the weight of edge $(i, j) \in E$

MC PROBLEM. Find partition of V into S and $V \setminus S$ s.t. the total weight of the edges joining S and $V \setminus S$ is maximized.

Max-cut

Given:

- $G = (V, E)$, …an undirected graph with $|V| = n$
- $w_{ij} = w_{ji} \geq 0$ …the weight of edge $(i, j) \in E$

MC PROBLEM. Find partition of V into S and $V \setminus S$ s.t. the total weight of the edges joining S and $V \setminus S$ is maximized.

$$x_j := \begin{cases} 1 & \text{for } j \in S \\ -1 & \text{for } j \in V \setminus S \end{cases}$$

$$(\text{MC}) \qquad \begin{array}{ll} \max & \frac{1}{4} \sum_{i,j=1}^{n} w_{ij}(1 - x_i x_j) \\ \text{s.t.} & x_j \in \{-1, 1\}, \quad j = 1, \ldots, n. \end{array}$$

- NP-hard problem

Goemans-Williamson

Let $Y = xx^{\mathrm{T}}$

$$
(\mathrm{MC}) \qquad \begin{array}{ll} \max & \frac{1}{4}\sum_{i,j=1}^{n} w_{ij}(1 - Y_{ij}) \\ \text{s.t.} & x_j \in \{-1, 1\}, \quad j = 1, \ldots, n \\ & Y = xx^{\mathrm{T}} \end{array}
$$

Goemans-Williamson

Let $Y = xx^{\mathrm{T}}$

$$
\text{(MC)} \quad \begin{array}{ll} \max & \frac{1}{4}\sum_{i,j=1}^{n} w_{ij}(1 - Y_{ij}) \\ \text{s.t.} & x_j \in \{-1, 1\}, \quad j = 1, \ldots, n \\ & Y = xx^{\mathrm{T}} \end{array}
$$

$$
\text{(MC)} \quad \begin{array}{ll} \max & \frac{1}{4}\sum_{i,j=1}^{n} w_{ij}(1 - Y_{ij}) \\ \text{s.t.} & Y_{jj} = 1, \quad j = 1, \ldots, n \\ & Y \succeq 0, \ \mathrm{rank}(Y) = 1 \end{array}
$$

Goemans-Williamson

Let $Y = xx^{\mathrm{T}}$

$$
(\mathrm{MC}) \quad \begin{aligned} \max \quad & \tfrac{1}{4}\textstyle\sum_{i,j=1}^{n} w_{ij}(1 - Y_{ij}) \\ \text{s.t.} \quad & x_j \in \{-1, 1\}, \quad j = 1, \dots, n \\ & Y = xx^{\mathrm{T}} \end{aligned}
$$

$$
(\mathrm{MC}) \quad \begin{aligned} \max \quad & \tfrac{1}{4}\textstyle\sum_{i,j=1}^{n} w_{ij}(1 - Y_{ij}) \\ \text{s.t.} \quad & Y_{jj} = 1, \quad j = 1, \dots, n \\ & Y \succeq 0, \ \mathrm{rank}(Y) = 1 \end{aligned}
$$

$\Rrightarrow$ **relax** the rank one constraint

$$
(\mathrm{SDP}_{\mathrm{MC}}) \quad \begin{aligned} \max \quad & \tfrac{1}{4}\textstyle\sum_{i,j=1}^{n} w_{ij}(1 - Y_{ij}) \\ \text{s.t.} \quad & Y_{jj} = 1, \quad j = 1, \dots, n \\ & Y \succeq 0 \end{aligned}
$$

$\Rrightarrow$ Goemans, Williamson (1995): this relaxation has an error $\leq 13.82\%$

Max-cut remarks

- strengthened $\mathrm{SDP}_{\mathrm{MC}}$ by adding $4\binom{n}{3}$ triangle constraints:

$$\mathrm{MET} \quad \begin{array}{rcl} y_{ij} + y_{ik} + y_{jk} & \geq & -1 \\ y_{ij} - y_{ik} - y_{jk} & \geq & -1 \\ -y_{ij} + y_{ik} - y_{jk} & \geq & -1 \\ -y_{ij} - y_{ik} + y_{jk} & \geq & -1, \ \forall i < j < k \end{array}$$

- the resulting SDP is difficult to solve with IPM when $n > 150$

- The bundle method computes nearly optimal solution for $n \leq 2000$:
  Fischer, Gruber, Rendl, and Sotirov. Computational Experience with a Bundle Approach for Semidefinite Cutting Plane Relaxations of Max-Cut and Equipartition, *Math. Program B*, 105(2-3):451-469, 2006.

- Branch and bound, SDP based solver for Max-cut:
  Rinaldi, Rendl and Wiegele. Biq Mac Solver - Binary quadratic and Max cut Solver, `http://biqmac.uni-klu.ac.at/`

THE GRAPH PARTITION PROBLEM . . .

The Graph Partition Problem

- $G = (V, E)$ ... an undirected graph
  - V ... vertex set, $|V| = n$
  - E ... edge set

THE k-PARTITION PROBLEM:
Find a partition of V into k subsets $S_1, \ldots, S_k$ of given sizes $m_1 \geq \ldots \geq m_k$, s.t. the total weight of edges joining different S_i is minimized.

- when $m_i = \frac{|V|}{k}$, $\forall i \rightsquigarrow$ the **graph equipartition problem**
- when $k = 2 \rightsquigarrow$ the **bisection problem**
- GPP is NP-hard (Garey and Johnson, 1976)
- **applications**: VLSI design, parallel computing, floor planning, telecommunications, etc.

The k-partition problem

- A ...the adjacency matrix of G, $m := (m_1, \ldots, m_k)^{\mathrm{T}}$, u_n all-ones vector
- let $X = (x_{ij}) \in \mathbb{R}^{|V| \times k}$

$$x_{ij} := \begin{cases} 1, & \text{if node } i \in S_j \\ 0, & \text{if node } i \notin S_j \end{cases}$$

- $\mathcal{P}_k := \left\{ X \in \mathbb{R}^{n \times k} : \ Xu_k = u_n, \ X^{\mathrm{T}} u_n = m, \ x_{ij} \in \{0, 1\} \right\}$

The k-partition problem

- A ...the adjacency matrix of G, $m := (m_1, \dots, m_k)^{\mathrm{T}}$, u_n all-ones vector
- let $X = (x_{ij}) \in \mathbb{R}^{|V| \times k}$

$$x_{ij} := \begin{cases} 1, & \text{if node } i \in S_j \\ 0, & \text{if node } i \notin S_j \end{cases}$$

- $\mathcal{P}_k := \left\{ X \in \mathbb{R}^{n \times k} : \; Xu_k = u_n, \; X^{\mathrm{T}} u_n = m, \; x_{ij} \in \{0, 1\} \right\}$

For $X \in \mathcal{P}_k$:

- $\frac{1}{2} \operatorname{tr}(X^{\mathrm{T}} A X) = \sum_j$ weight of edges within S_j:

$$\mathbf{w}(\mathbf{E}_{\mathrm{cut}}) = \frac{1}{2} \operatorname{tr}(X^{\mathrm{T}} \mathrm{Diag}(Au_n) X - X^{\mathrm{T}} A X) = \frac{1}{2} \operatorname{tr}(X^{\mathrm{T}} L X),$$

where $L := \mathrm{Diag}(Au_n) - A$ is the Laplacian matrix of G

The Graph Partition Problem

THE TRACE FORMULATION:

$$
\text{(GPP)} \quad \begin{array}{ll} \min & \frac{1}{2}\mathrm{trace}(X^{\mathrm{T}}LX) \\ \text{s.t.} & Xu_k = u_n \\ & X^{\mathrm{T}}u_n = m \\ & x_{ij} \in \{0,1\} \end{array}
$$

SDP for GPP

- linearize the objective (matrix lifting): $\mathrm{trace}(\boldsymbol{L}\mathbf{X}\mathbf{X}^{\mathrm{T}}) \rightsquigarrow \mathrm{trace}(\boldsymbol{L}\mathbf{Y})$

$$Y \in \mathrm{conv}\{\tilde{Y} : \exists X \in \mathcal{P}_k \text{ s.t. } \tilde{Y} = XX^{\mathrm{T}}\} \Rightarrow kY - J_n \succeq 0.$$

SDP for GPP

- linearize the objective (matrix lifting): $\mathrm{trace}(L\mathbf{X}\mathbf{X}^{\mathrm{T}}) \rightsquigarrow \mathrm{trace}(L\mathbf{Y})$

$Y \in \mathrm{conv}\{\tilde{Y} : \exists X \in \mathcal{P}_k \text{ s.t. } \tilde{Y} = XX^{\mathrm{T}}\} \Rightarrow kY - J_n \succeq 0.$

$$(\mathrm{GPP}_{\mathrm{RS}}) \qquad \begin{aligned} \min \quad & \tfrac{1}{2}\,\mathrm{tr}(LY) \\ \text{s.t.} \quad & \mathrm{diag}(Y) = u_n \\ & \mathrm{tr}(JY) = \sum_{i=1}^{k} m_i^2 \\ & kY - J_n \succeq 0, \;\; Y \geq 0 \end{aligned}$$

Sotirov, An efficient SDP relaxation for the GPP, INFORMS J. Comput. (2014)

Improvements?

? HOW TO IMPROVE $\mathrm{GPP}_{\mathrm{RS}}$?

impose the linear inequalities:

- Δ constraints

$$y_{ij} + y_{ik} \leq 1 + y_{jk}, \quad \forall (i, j, k)$$

- independent set type of constraints

$$\sum_{i<j, i,j \in \mathcal{I}} y_{ij} \geq 1, \ \forall \ \mathcal{I} \text{ s.t. } |\mathcal{I}| = k + 1$$

$\Rightarrow$ there are $3\binom{n}{3}$ Δ, and $\binom{n}{k+1}$ independent set constraints

Some facts ...

for graphs with 100 vertices:

- the best known vector lifting relaxation is **hopeless**
- $\mathrm{GPP}_{\mathrm{RS}} + \Delta +$ independent set constraints computes bounds in about 3 hours
- $\mathrm{GPP}_{\mathrm{RS}}$ computes bounds in about 14 minutes

? CAN WE COMPUTE $\mathrm{GPP}_{\mathrm{RS}}$ MORE EFFICIENTLY ?

Some facts . . .

for graphs with 100 vertices:

- the best known vector lifting relaxation is **hopeless**
- $\mathrm{GPP}_{\mathrm{RS}} + \Delta +$ independent set constraints computes bounds in about 3 hours
- $\mathrm{GPP}_{\mathrm{RS}}$ computes bounds in about 14 minutes

? CAN WE COMPUTE $\mathrm{GPP}_{\mathrm{RS}}$ MORE EFFICIENTLY ?

YES

Symmetry and algebra

- matrix *-algebra: subspace of $\mathbb{R}^{n \times n}$ that is closed under matrix multiplication and transposition

Assumption: The data matrices of an SDP problem and I belong to a matrix *-algebra of dimension r, where $r \ll n^2$

Symmetry and algebra

- matrix *-algebra: subspace of $\mathbb{R}^{n\times n}$ that is closed under matrix multiplication and transposition

Assumption: The data matrices of an SDP problem and I belong to a
matrix *-algebra of dimension r, where $r \ll n^2$

Then, if the SDP relaxation has an optimal solution
$\Rightarrow$ then it has an optimal solution in the matrix *-algebra.

Schrijver, Goemans, Rendl, Parrilo, De Klerk, Pasechnik, Sotirov, ...

Symmetry and algebra

- matrix *-algebra: subspace of $\mathbb{R}^{n\times n}$ that is closed under matrix multiplication and transposition

Assumption: The data matrices of an SDP problem and I belong to a matrix *-algebra of dimension r, where $r \ll n^2$

Then, if the SDP relaxation has an optimal solution
$\Rightarrow$ then it has an optimal solution in the matrix *-algebra.

Schrijver, Goemans, Rendl, Parrilo, De Klerk, Pasechnik, Sotirov, . . .

- Coherent algebra with basis of 01-matrices (centralizer ring, for example):

(i) $A_i \in \{0,1\}^{n\times n}$, $A_i^{\mathrm{T}} \in \{A_1, \ldots, A_r\}$, $(i = 1, \ldots, r)$

(ii) $\sum_{i=1}^r A_i = J$, $\sum_{i\in\mathcal{I}} A_i = I$, $\mathcal{I} \subset \{1, \ldots, r\}$

(iii) For $i, j \in \{1, \ldots, r\}$, $\exists p_{ij}^h$ such that $A_iA_j = \sum_{h=1}^r p_{ij}^h A_h$.

Simplification – 'highly symmetric' graphs ...

$\Rightarrow Y = \sum_{i=1}^{r} \mathbf{z_i} A_i,$

$$
(\mathrm{GPP_m}) \quad
\begin{array}{ll}
\min & \frac{1}{2}\operatorname{tr}(AJ_n) - \frac{1}{2}\sum_{i=1}^{r} \mathbf{z_i} \operatorname{tr}(AA_i) \\
\text{s.t.} & \sum_{i\in\mathcal{I}} \mathbf{z_i} \operatorname{diag}(A_i) = u_n \\
& \sum_{i=1}^{r} \mathbf{z_i} \operatorname{tr}(JA_i) = \sum_{i=1}^{k} m_i^2 \\
& k\sum_{j=1}^{r} \mathbf{z_i} A_i - J_n \succeq 0, \quad \mathbf{z_i} \geq 0, \quad i = 1, \ldots, r.
\end{array}
$$

Simplification – 'highly symmetric' graphs ...

$$\Rightarrow Y = \sum_{i=1}^{r} \mathbf{z_i} A_i,$$

$$(\mathrm{GPP_m}) \quad \begin{array}{ll} \min & \frac{1}{2}\operatorname{tr}(AJ_n) - \frac{1}{2}\sum_{i=1}^{r} \mathbf{z_i} \operatorname{tr}(AA_i) \\ \text{s.t.} & \sum_{i\in\mathcal{I}} \mathbf{z_i} \operatorname{diag}(A_i) = u_n \\ & \sum_{i=1}^{r} \mathbf{z_i} \operatorname{tr}(JA_i) = \sum_{i=1}^{k} m_i^2 \\ & k\sum_{j=1}^{r} \mathbf{z_i} A_i - J_n \succeq 0, \quad \mathbf{z_i} \geq 0, \quad i = 1, \ldots, r. \end{array}$$

- LMI may be (block-)diagonalized
- exploit properties of A_i to aggregate Δ and independent set constraints

  $\Rightarrow$ extend the approach from:

  M.X. Goemans, F. Rendl. Semidefinite Programs and Association Schemes. *Computing*, 63(4):331–340, 1999.

On aggregating constraints ...

- for a given (a, b, c) consider the Δ constraint

$$y_{ab} + y_{ac} \leq 1 + y_{bc}$$

- if $(A_i)_{ab} = 1$, $(A_h)_{ac} = 1$, $(A_j)_{bc} = 1 \leftarrow$ type (i, j, h) constraint

On aggregating constraints ...

- for a given (a, b, c) consider the Δ constraint

$$y_{ab} + y_{ac} \leq 1 + y_{bc}$$

- if $(A_i)_{ab} = 1$, $(A_h)_{ac} = 1$, $(A_j)_{bc} = 1 \leftarrow$ type (i, j, h) constraint

- summing all constraints of type $(i, j, h) \rightarrow$ aggregated Δ constraint:

$$p^i_{hj'} \operatorname{tr} A_i Y + p^h_{ij} \operatorname{tr} A_h Y \leq p^i_{hj'} \operatorname{tr} A_i J + p^j_{i'h} \operatorname{tr} A_j Y,$$

On aggregating constraints ...

- for a given (a, b, c) consider the Δ constraint

$$y_{ab} + y_{ac} \leq 1 + y_{bc}$$

- if $(A_i)_{ab} = 1$, $(A_h)_{ac} = 1$, $(A_j)_{bc} = 1 \leftarrow$ type (i, j, h) constraint

- summing all constraints of type $(i, j, h) \rightarrow$ aggregated Δ constraint:

$$p^i_{hj'} \operatorname{tr} A_i Y + p^h_{ij} \operatorname{tr} A_h Y \leq p^i_{hj'} \operatorname{tr} A_i J + p^j_{i'h} \operatorname{tr} A_j Y,$$

- $\sharp$ of aggregated Δ constraints is bounded by r^3
- similar approach applies to independent set constraints when $k = 2$

Strongly regular graphs

Example. Strongly regular graph

- n vertices, κ the *valency* of the graph
- A has exactly two eigenvalues $r \geq 0$ and $s < 0$ associated with eigenvectors $\perp u_n$
- A belongs to the *-algebra spanned by $\{I, A, J - A - I\}$

$$\Rightarrow Y = I + z_1 A + z_2 (J - A - I)$$

Strongly regular graphs

Example. Strongly regular graph

- n vertices, κ the *valency* of the graph
- A has exactly two eigenvalues $r \geq 0$ and $s < 0$ associated with eigenvectors $\perp u_n$
- A belongs to the *-algebra spanned by $\{I, A, J - A - I\}$

$\Rightarrow Y = I + z_1 A + z_2 (J - A - I)$

$$
(\mathrm{GPP_m}) \quad
\begin{array}{ll}
\min & \frac{1}{2}\kappa n(1 - z_1) \\
\text{s.t.} & \kappa z_1 + (n - \kappa - 1) z_2 = \frac{1}{n}\sum_{i=1}^{k} m_i^2 - 1 \\
& 1 + r z_1 - (r+1) z_2 \geq 0 \\
& 1 + s z_1 - (s+1) z_2 \geq 0 \\
& z_1, z_2 \geq 0
\end{array}
$$

Strongly regular graphs ....

THEOREM.
Let $G = (V, E)$ be a SRG with eigenvalues κ, r, s.
Let $m_i \in \mathbf{N}$, $i = 1, \ldots, k$ s.t. $\sum_{j=1}^{k} m_j = n$.

Then the SDP (lower) bound for the minimum k-partition is

$$\max\left\{\tfrac{\kappa-r}{n}\sum_{i<j} m_i m_j,\ \tfrac{1}{2}\left(n(\kappa+1)-\sum_i m_i^2\right)\right\}$$

Similarly, the SDP (upper) bound for the maximum k-partition is

$$\min\left\{\tfrac{\kappa-s}{n}\sum_{i<j} m_i m_j,\ \tfrac{1}{2}\kappa n\right\}.$$

Strongly regular graphs ....

THEOREM.
Let $G = (V, E)$ be a SRG with eigenvalues κ, r, s.
Let $m_i \in \mathbf{N}$, $i = 1, \ldots, k$ s.t. $\sum_{j=1}^{k} m_j = n$.

Then the SDP (lower) bound for the minimum k-partition is

$$\max\left\{\frac{\kappa - r}{n}\sum_{i<j} m_i m_j,\ \frac{1}{2}\left(n(\kappa+1) - \sum_i m_i^2\right)\right\}$$

Similarly, the SDP (upper) bound for the maximum k-partition is

$$\min\left\{\frac{\kappa - s}{n}\sum_{i<j} m_i m_j,\ \frac{1}{2}\kappa n\right\}.$$

- this is an extension of the result for the equipartition:

  De Klerk, Pasechnik, Sotirov, Dobre: On SDP relaxations of maximum k-section, *Math. Program. Ser. B*, 136(2):253-278, 2012.

Adding constraints

- after aggregating Δ constraints:

$$\begin{aligned} z_1 &\leq 1 \\ z_2 &\leq 1 \\ 2z_1 - z_2 &\leq 1 \\ -z_1 + 2z_2 &\leq 1 \end{aligned}$$

For SRG with $n > 5$ the Δ constraints are redundant in $\mathrm{GPP_m}$.

However, the independent set constraints improve $\mathrm{GPP_m}$.

The Laplacian algebra

- closed form expression for the GPP for 'any' graph
- $L = \mathrm{Diag}(Au_n) - A$, the Laplacian matrix of G
- $\mathcal{L} := \mathrm{span}\{F_0, \ldots, F_d\}$, the Laplacian algebra of G
- $F_i = U_i U_i^{\mathrm{T}}$ (eigenspace decomposition, $LU_i = \lambda_i U_i$)
  - $F_i F_j = \delta_{ij} F_i$ for $i \neq j$
  - $\sum_{i=0}^{d} F_i = I$
  - $F_i = F_i^{\mathrm{T}}$, $\forall i$
  - $\mathrm{tr}(F_i) = f_i$ . . . the multiplicity of i-th eigenvalue of L

Simplification – any graph . . .

- **relax** $\mathrm{diag}(Y) = u_n \rightsquigarrow \mathrm{tr}(Y) = n$
- **remove** nonnegativity constraint

$$(\mathrm{GPP}_{\mathrm{eig}}) \qquad \begin{aligned} \min \quad & \tfrac{1}{2}\,\mathrm{tr}\,LY \\ \text{s.t.} \quad & \mathrm{tr}(Y) = n \\ & \mathrm{tr}(JY) = \sum_{i=1}^{k} m_i^2 \\ & kY - J_n \succeq 0 \end{aligned}$$

Simplification – any graph ...

- **relax** $\mathrm{diag}(Y) = u_n \rightsquigarrow \mathrm{tr}(Y) = n$
- **remove** nonnegativity constraint

$$
(\mathrm{GPP}_{\mathrm{eig}}) \qquad \begin{array}{rl} \min & \frac{1}{2}\,\mathrm{tr}\, LY \\ \text{s.t.} & \mathrm{tr}(Y) = n \\ & \mathrm{tr}(JY) = \sum_{i=1}^{k} m_i^2 \\ & kY - J_n \succeq 0 \end{array}
$$

- $Y = \sum_{i=0}^{d} \mathbf{y_i} F_i, \quad \mathbf{y_i} \in \mathbb{R}\ (i = 0, \dots, d)$

$$
\mathrm{tr}(LY) = \mathrm{tr}(\sum_{j=0}^{d} \lambda_j F_j (\sum_{i=0}^{d} \mathbf{y_i} F_i)) = \sum_{i=0}^{d} \lambda_i f_i \mathbf{y_i}
$$

where $0 = \lambda_0 \leq \ldots \leq \lambda_d$ distinct eigenvalues of L, ...

Eigenvalue bounds

Theorem

Let $G = (V, E)$ be a graph, m_i, $i = 1, \ldots, k$ s.t. $\sum_{j=1}^{k} m_j = n$. Then the $\mathrm{GPP}_{\mathrm{eig}}$ bound for the minimum k-partition of G equals

$$\frac{\lambda_1}{n} \sum_{i<j} m_i m_j,$$

and the bound $\mathrm{GPP}_{\mathrm{eig}}$ for the maximum k-partition of G equals

$$\frac{\lambda_d}{n} \sum_{i<j} m_i m_j.$$

Eigenvalue bounds

THEOREM

Let $G = (V, E)$ be a graph, m_i, $i = 1, \ldots, k$ s.t. $\sum_{j=1}^{k} m_j = n$. Then the GPP_{eig} bound for the minimum k-partition of G equals

$$\frac{\lambda_1}{n} \sum_{i<j} m_i m_j,$$

and the bound GPP_{eig} for the maximum k-partition of G equals

$$\frac{\lambda_d}{n} \sum_{i<j} m_i m_j.$$

- Other known closed form expression only for the minimum k-partition when $k = 2, 3$:

  J. Falkner, F. Rendl, and H. Wolkowicz. A computational study of graph partitioning. Math. Program., 66:211–239, 1994.

Computational times for presented bounds

- exploit symmetry if available

G	n	m	time, no symmetry	r_{aut}	time
grid graph	100	(50,25,25)	799.2	1275	**3.4**

Table: Computational time (s.) to solve $\mathrm{GPP}_{\mathrm{RS}}$

- computational time to solve $\mathrm{GPP}_{\mathrm{RS}} + \Delta$ constraints, with $n = 100$:
  - without symmetry, about 2 hours
  - aggregating constraints, if possible, a few seconds

Quality of the presented bounds

G	n	k	$\mathrm{GPP}_{\mathrm{eig}}$	$\mathrm{GPP}_{\mathrm{RS}}$	r_{comb}	time	r_{aut}	time
Chang3	28	7	96	126	3	–	14	0.23
SRG$(64, 18)_{30}$	64	8	448	448	3	–	90	0.61
Doob	64	8	112	160	4	0.34	8	0.41
SRG$(64, 18)_e$	64	4	384	384	3	–	–	14.33
$(45, 12, 3)$-design	90	9	360	360	4	0.40	2074	4.56

G	n	k	$\mathrm{GPP}_{\mathrm{eig}}$	$\mathrm{GPP}_{\mathrm{RS}}$	$\mathrm{GPP}_{\mathrm{RS}} + \Delta$
Desargues	20	2	5	5	6
Foster	90	5	20	23	31
Biggs-Smith	102	3	15	15	23

Table: Lower bounds for the min k-partition.

- each computation in the last two columns of the last table < 1 s.

The max-k-cut

- $G = (V, E)$ ... an undirected graph
  - V ... vertex set, $|V| = n$
  - E ... edge set

The max-k-cut problem

Find a partition of V into into at most k subsets such that the total weight of edges joining different sets is maximized.

the max-k-cut ...

- is NP-hard
- $k = 2 \rightsquigarrow$ the **max-cut**

An SDP relaxation

- A . . . the adjacency matrix of G
- $L := \mathrm{Diag}(Au_n) - A$ is the Laplacian matrix of G

$$(k - \mathrm{MC}) \qquad \begin{array}{ll} \max & \frac{1}{2}\operatorname{tr}(LY) \\ \text{s.t.} & \operatorname{diag}(Y) = u_n \\ & kY - J_n \succeq 0, \;\; Y \geq 0 \end{array}$$

where J_n (resp. u_n) is all-ones matrix (resp. vector)

- restriction on sizes of the parts $\rightarrow$ GPP
- Δ and independent set constraints can be added
- $(k - \mathrm{MC})$ is equivalent to the relaxation from:
  Frieze, Jerrum. Improved approximation algorithms for max-k-cut and max bisection. *Algorithmica* 18:67-81, 1997.

Eigenvalue bounds: the max-k-cut

- **relax** $\mathrm{diag}(Y) = u_n \rightsquigarrow \mathrm{tr}(Y) = n$
- **remove** nonnegativity constraint

Theorem

Let $G = (V, E)$ be a graph on n vertices and k an integer $k \geq 2$.
The eigenvalue (upper) bound for the max-k-cut problem is:

$$\frac{n(k-1)}{2k} \lambda_{\max}(L).$$

Eigenvalue bounds: the max-k-cut

- **relax** $\text{diag}(Y) = u_n \rightsquigarrow \text{tr}(Y) = n$
- **remove** nonnegativity constraint

Theorem

Let $G = (V, E)$ be a graph on n vertices and k an integer $k \geq 2$.
The eigenvalue (upper) bound for the max-k-cut problem is:

$$\frac{n(k-1)}{2k}\lambda_{\max}(L).$$

- for $k = 2$ this result coincides with:

  Mohar and Poljak. Eigenvalues and the max-cut problem.
  Czechoslovak Mathematical Journal, 40:343-352, 1990.

- there are few other eigenvalue bounds for the max-k-cut when $k > 2$ (Nikiforov)

The chromatic number

The chromatic number of a graph

- A coloring of a graph is an assignment of colors to the vertices of G s.t. no two adjacent vertices have the same color.
- The smallest number of colors needed to color G is called its chromatic number $\chi(G)$.

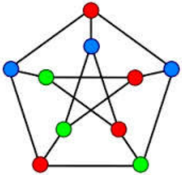


Figure: Petersen graph, $\chi(G) = 3$

The chromatic number

The chromatic number of a graph

- A coloring of a graph is an assignment of colors to the vertices of G s.t. no two adjacent vertices have the same color.
- The smallest number of colors needed to color G is called its chromatic number $\chi(G)$.

$\Rrightarrow$ A coloring with k colors is the same as a partition V into k independent sets.

For a given graph $G = (V, E)$ and integer k,

$$\text{if} \quad \text{max-}k\text{-cut} < |E| \quad \text{then} \quad \chi(G) \geq k + 1.$$

$\Rrightarrow$ The eigenvalue bound for the max-k-cut $\rightsquigarrow$ a bound on $\chi(G)$

Eigenvalue bound for the chromatic number

Theorem

Let $G = (V, E)$ be a graph with Laplacian matrix L. Then

$$\chi(G) \geq 1 + \frac{2|E|}{n\lambda_{\max}(L) - 2|E|}$$

Eigenvalue bound for the chromatic number

Theorem

Let $G = (V, E)$ be a graph with Laplacian matrix L. Then

$$\chi(G) \geq 1 + \frac{2|E|}{n\lambda_{\max}(L) - 2|E|}$$

- Hoffman bound, 1970: $\chi(G) \geq 1 - \frac{\theta_{\max}(A)}{\theta_{\min}(A)}$, where $\theta_{\max}(A)$ and $\theta_{\min}(A)$ are largest and smallest eigenvalue of adjacency matrix A.
- For regular graphs, these two bounds coincide. Otherwise, they are incomparable.
- For the complete graph on 100 vertices minus an edge, our bound is 99 $(= \chi(G))$ while the Hoffman bound is 51.

Walk-regular graphs

Walk-regular graphs

A graph with adjacency matrix A is called walk-regular if A^ℓ has constant diagonal for every nonnegative integer ℓ.

The class of walk-regular graphs contains:

- vertex-transitive graphs
- distance-regular graphs (including strongly regular graphs)
- graphs in an association scheme

Walk-regular graphs:

- are regular graphs
- **all** matrices in $\mathcal{L}$ of a walk-regular graph have constant diagonal

Max-k-cut for walk-regular graphs

$\Rightarrow$ the optimum correcting vector d in

$$(\clubsuit) \qquad \min_{d^{\mathrm{T}} u_n = 0} \frac{n(k-1)}{2k} \lambda_{\max}(L + \mathrm{Diag}(d))$$

equals the zero vector.

Max-k-cut for walk-regular graphs

$\Rrightarrow$ the optimum correcting vector d in

$$(\clubsuit) \qquad \min_{d^{\mathrm{T}} u_n = 0} \frac{n(k-1)}{2k} \lambda_{\max}(L + \mathrm{Diag}(d))$$

equals the zero vector.

Theorem

Let G be a walk-regular graph on n vertices and let k be an integer $k \geq 2$.

Then the eigenvalue bound for the max-k-cut equals the bound $(\clubsuit)$.

For $k = 2$ the eigenvalue bound equals the optimal value of the SDP rel. (k–MC).

- Goemans and Rendl (1999) proved the latter result for the max-cut problem for graphs in an association scheme.

Strongly regular graph

Theorem

Let $G = (V, E)$ be a SRG with eigenvalues κ, r, s.
Then the SDP bound (k-MC) for the max-k-cut of G is given by

$$\min\left\{\frac{n(k-1)}{2k}(\kappa - s),\ \frac{1}{2}\kappa n\right\}.$$

- For SRG with $n > 5$ the (aggregated) Δ constraints are redundant in $(k - MC)$.
- the independent set constraints improve $(k - MC)$

Hamming Graphs

Hamming graph $H(d, q, j)$ ($j = 0, \dots, d$)

- Vertex set S^d, where S is a set of size q
- Vertices are adjacent if they differ in j coordinates

Conjecture

Let $j \geq d - \frac{d-1}{q}$, with j even if $q = 2$. Then $K_j(0) - K_j(1)$ (Kravchuk) is the largest Laplacian eigenvalue of $H(d, q, j)$.

Hamming Graphs

Hamming graph $H(d, q, j)$ ($j = 0, \ldots, d$)

- Vertex set S^d, where S is a set of size q
- Vertices are adjacent if they differ in j coordinates

Conjecture

Let $j \geq d - \frac{d-1}{q}$, with j even if $q = 2$. Then $K_j(0) - K_j(1)$ (Kravchuk) is the largest Laplacian eigenvalue of $H(d, q, j)$.

Theorem

Let $k \leq q$, $j \geq d - \frac{d-1}{q}$, with j even if $q = 2$, and consider $H(d, q, j)$.
If the conjecture is true, then:

- for the max-k-cut problem, the eigenvalue and $(k - MC)$ bound are equal.
- the optimal value of the max-q-cut equals the eigenvalue bound.

Hamming Graphs

Hamming graph $H(d,q,j)$ ($j=0,\ldots,d$)

- Vertex set S^d, where S is a set of size q
- Vertices are adjacent if they differ in j coordinates

Conjecture

Let $j \geq d - \frac{d-1}{q}$, with j even if $q=2$. Then $K_j(0)-K_j(1)$ (Kravchuk) is the largest Laplacian eigenvalue of $H(d,q,j)$.

Theorem

Let $k \leq q$, $j \geq d - \frac{d-1}{q}$, with j even if $q=2$, and consider $H(d,q,j)$.
If the conjecture is true, then:

- for the max-k-cut problem, the eigenvalue and $(k-MC)$ bound are equal.
- the optimal value of the max-q-cut equals the eigenvalue bound.

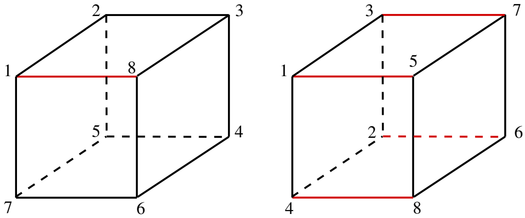
The conjecture was recently proven by Brouwer, Cioabă, Ihringer, and McGinnis (JCTB, 201?)

The bandwidth issue

"We both work at home, so we compete for bandwidth, not closet space."

The Bandwidth Problem for graphs

$$\sigma(G) := \min\left\{ \max_{(i,j)\in E} |\phi(i)-\phi(j)|;\ \phi : V \to \{1,\dots,n\} \right\}$$

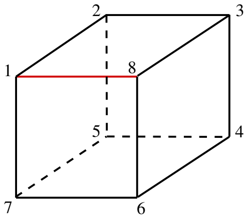


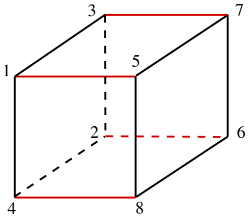
$$\Longleftarrow\!\!\Longrightarrow$$

find a PERMUTATION P such that in PAP^{T} ALL nonzero entries are as close as possible to the main diagonal

The Bandwidth Problem for graphs

$$\sigma(G) := \min \left\{ \max_{(i,j)\in E} |\phi(i) - \phi(j)|;\ \phi : V \to \{1, \ldots, n\} \right\}$$





$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{pmatrix}$$

Applications . . .

The bandwidth problem . . .

- originated in the 1950s from sparse matrix computations
- NP-hard (Papadimitriou (1976))
- engineering applications
  - efficient storage and processing
  - minimizing distortion in the multi-channel transmission

A quadratic assignment formulation of the bandwidth

$\Rrightarrow$ "natural" problem formulation

- Fix k ...
- define $B = (b_{ij})$:

$$b_{ij} := \begin{cases} 1 & \text{for } |i-j| > k \\ 0 & \text{otherwise} \end{cases}$$

- the bandwidth related to the QAP:

$$\mu^* = \min_{P \in \Pi_n} \operatorname{tr}(APBP^{\mathrm{T}}),$$

where Π_n is the set of **permutation** matrices

$$\text{if} \quad \mu^* > 0 \quad \Rightarrow \quad \sigma(G) > k$$

The QAP based bound

$$
\left.
\begin{array}{rl}
\alpha_{QAP} := \min & \operatorname{tr}(B \otimes A)Y \\
\text{s.t.} & \operatorname{tr}(I \otimes E_{jj})Y = 1, \quad \operatorname{tr}(E_{jj} \otimes I)Y = 1 \quad \forall j \\
& \operatorname{tr}(I \otimes (J - I)) + (J - I) \otimes I)Y = 0 \\
& \operatorname{tr}(JY) = n^2 \\
& Y \geq 0, \quad Y \succeq 0
\end{array}
\right\} \qquad (\Diamond)
$$

- $E_{jj} = e_j e_j^T$
- I is the identity matrix
- J is all-ones matrix
- QAP SDP formulation by Povh and Rendl (2009); Zhao, Karisch, Rendl, and Wolkowicz (1998)

The QAP based bound

$$\left.\begin{aligned} \alpha_{QAP} := \min \quad & \operatorname{tr}(B \otimes A)Y \\ \text{s.t.} \quad & \operatorname{tr}(I \otimes E_{jj})Y = 1, \quad \operatorname{tr}(E_{jj} \otimes I)Y = 1 \quad \forall j \\ & \operatorname{tr}(I \otimes (J - I)) + (J - I) \otimes I)Y = 0 \\ & \operatorname{tr}(JY) = n^2 \\ & Y \geq 0, \quad Y \succeq 0 \end{aligned}\right\} \qquad (\diamondsuit)$$

- $E_{jj} = e_j e_j^T$
- I is the identity matrix
- J is all-ones matrix
- QAP SDP formulation by Povh and Rendl (2009); Zhao, Karisch, Rendl, and Wolkowicz (1998)
- since $\operatorname{aut}(B) = \{P \in \Pi_n : \ PBP^{\mathrm{T}} = B\}$ has **order** only 2

$\rightsquigarrow$ look for **other** approaches

The min-cut problem

- $S_1, S_2, S_3 \subseteq V$
- $|S_i| = m_i$ for $i = 1, 2, 3, \quad \sum_i m_i = n$

The min-cut problem is:

$$
(\mathrm{MC}) \qquad \begin{array}{rll} \mathrm{OPT}_{\mathrm{MC}} := & \min & \displaystyle\sum_{i \in S_1, j \in S_2} a_{ij} \\ & \text{s.t.} & (S_1, S_2, S_3) \ \text{ partitions } \ V \end{array}
$$

where $A = (a_{ij})$ is the adjacency matrix.

? relation to the bandwidth problem ?

The min-cut and bandwidth problem

- The cube, $m = (m_1, m_2, m_3) = (2, 3, 3)$

$$
\begin{array}{cccccccc}
\multicolumn{2}{c}{m_1} & \multicolumn{3}{c}{m_3} & \multicolumn{3}{c}{m_2} \\
\end{array}
$$
$$
\left(\begin{array}{cc|ccc|ccc}
0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\
\hline
1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
\hline
0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 0 & 0
\end{array}\right)
$$

$$\text{if} \quad \mathrm{OPT}_{\mathrm{MC}} > 0 \quad \Rightarrow \quad \sigma(G) \geq m_3 + 1$$

The min-cut and bandwidth problem

- The cube, $m = (m_1, m_2, m_3) = (2, 3, 3)$

$$
\begin{array}{c}
\begin{array}{ccccc} \quad m_1 & \qquad & m_3 & \qquad\qquad & m_2 \end{array} \\
\left(\begin{array}{cc|ccc|ccc}
0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\
\hline
1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\
\hline
0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\
0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 1 & 1 & 0 & 0
\end{array}\right)
\end{array}
$$

$$\text{if} \quad \mathrm{OPT}_{\mathrm{MC}} > 0 \quad \Rightarrow \quad \sigma(G) \geq m_3 + 1$$

generalized bound (Povh-Rendl (2007), EvD-Sotirov):

If for some $m = (m_1, m_2, m_3)$ it holds that $\mathrm{OPT}_{\mathrm{MC}} \geq \alpha > 0$, then
$$\sigma(G) \geq m_3 + \left\lceil -\frac{1}{2} + \sqrt{2\alpha + \frac{1}{4}} \right\rceil$$

Eigenvalue based bound for the min-cut

- Helmberg, Rendl, Mohar, Poljak (1995), derive the mc bound:

$$\alpha_{\mathrm{L}} = -\frac{1}{2}(\mu_2\lambda_2 + \mu_1\lambda_n),$$

  where
  - λ_2 (λ_n) ... the second smallest (largest resp.) Laplacian eigenvalue of G
  - μ_1 and μ_2 are constants depending on $m = (m_1, m_2, m_3)$

- α_{L} is the closed form solution of a minimization problem over

$$\{X \in \mathbb{R}^{n\times 3} : X^{\mathrm{T}}X = \mathrm{Diag}(m), Xu_3 = u_n, X^{\mathrm{T}}u_n = m\}$$

The QAP based bound for the min-cut

- the min-cut can be formulated as the QAP

$$\min_{X \in \Pi_n} \operatorname{tr}(AXBX^{\mathrm{T}}),$$

where A is the adjacency matrix of G and

$$B := \begin{pmatrix} 0_{m_1 \times m_1} & 0_{m_1 \times m_3} & J_{m_1 \times m_2} \\ 0_{m_3 \times m_1} & 0_{m_3 \times m_3} & 0_{m_3 \times m_2} \\ J_{m_2 \times m_1} & 0_{m_2 \times m_3} & 0_{m_2 \times m_2} \end{pmatrix}$$

The QAP based bound for the min-cut

- the min-cut can be formulated as the QAP

$$\min_{X \in \Pi_n} \operatorname{tr}(AXBX^{\mathrm{T}}),$$

where A is the adjacency matrix of G and

$$B := \begin{pmatrix} 0_{m_1\times m_1} & 0_{m_1\times m_3} & J_{m_1\times m_2} \\ 0_{m_3\times m_1} & 0_{m_3\times m_3} & 0_{m_3\times m_2} \\ J_{m_2\times m_1} & 0_{m_2\times m_3} & 0_{m_2\times m_2} \end{pmatrix}$$

- B generates a coherent algebra of rank 12.

On solving ($\diamondsuit$) relaxation

$$
\begin{aligned}
\min \quad & \tfrac{1}{2}\operatorname{tr} A(X_3 + X_5) \\
\text{s.t.} \quad & X_1 + X_6 + X_{11} = I_{n-2} \\
& \sum_{i=1}^{12} X_i = J_{n-2}, \\
& \operatorname{tr}(JX_i) = p_i,\ X_i \geq 0,\ i = 1, \dots, 12, \\
& \sum_{i=1}^{12} p_i^{-1} B_i \otimes X_i \succeq 0 \\
& X_3 = X_5^{\mathrm{T}}, X_4 = X_9^{\mathrm{T}}, X_8 = X_{11}^{\mathrm{T}}, \\
& X_1, X_2, X_6, X_7, X_{11}, X_{12} \in \mathcal{S}_{n-2},
\end{aligned}
$$

where

- p_i $(i = 1, \dots, 12)$ are given constants related $m = (m_1, m_2, m_3)$
- this reduction was introduced by De Klerk and Sotirov (2010), see also: E. de Klerk, F.M. de Oliveira Filho, and D.V. Pasechnik. Relaxations of combinatorial problems via association schemes, in *Handbook of Semidefinite, Cone and Polynomial Optimization*, Miguel Anjos and Jean Lasserre (eds.), pp. 171–200, Springer, 2012.

The QAP based bound for the min-cut

Theorem. Let G be an undirected graph with n vertices and adjacency matrix A, and $m = (m_1, m_2, m_3)$, $\sum_i m_1 = n$. Then,

$$\alpha_{QAP} \geq \alpha_L.$$

? How can we further improve the lower bound for the min-cut ?

Bounds for the min-cut

New bound for the min-cut

- assume: G_A is edge transitive
- the graph with adjacency matrix B is edge transitive

$\Rrightarrow$ one can fix arbitrary edge in B and compute a lower bound for the original QAP from the SDP relaxation of the "reduced" QAP

New bound for the min-cut

- assume: G_A is edge transitive
- the graph with adjacency matrix B is edge transitive

$\Rightarrow$ one can fix arbitrary edge in B and compute a lower bound for the original QAP from the SDP relaxation of the "reduced" QAP

Theorem. Let G_A be an undirected graph with adjacency matrix A. Suppose for simplicity that $\mathrm{aut}(G_A)$ is transitive on both edges and non-edges.

Then for any fixed edge (s_1, s_2) in G_B, and any fixed edge (r_1, r_2) and non-edge (q_1, q_2) in the graph G_A one has

$$\min_{X \in \Pi_n} \mathrm{tr}(AXBX^{\mathrm{T}}) = \min\{\min_{Z \in \Pi_n} \mathrm{tr}(AZBZ^{\mathrm{T}}), \min_{Y \in \Pi_n} \mathrm{tr}(AYBY^{\mathrm{T}})\}$$

where $Z_{r_1,s_1} = 1$, $Z_{r_2,s_2} = 1$ and $Y_{q_1,s_1} = 1$, $Y_{q_2,s_2} = 1$.

New bound for the min-cut

- assume: G_A is edge transitive
- the graph with adjacency matrix B is edge transitive

$\Rrightarrow$ one can fix arbitrary edge in B and compute a lower bound for the original QAP from the SDP relaxation of the "reduced" QAP

Theorem. Let G_A be an undirected graph with adjacency matrix A. Suppose for simplicity that $\mathrm{aut}(G_A)$ is transitive on both edges and non-edges.

Then for any fixed edge (s_1, s_2) in G_B, and any fixed edge (r_1, r_2) and non-edge (q_1, q_2) in the graph G_A one has

$$\min_{X \in \Pi_n} \mathrm{tr}(AXBX^{\mathrm{T}}) = \min\{\min_{Z \in \Pi_n} \mathrm{tr}(AZBZ^{\mathrm{T}}), \min_{Y \in \Pi_n} \mathrm{tr}(AYBY^{\mathrm{T}})\}$$

where $Z_{r_1, s_1} = 1$, $Z_{r_2, s_2} = 1$ and $Y_{q_1, s_1} = 1$, $Y_{q_2, s_2} = 1$.

- this can be extended to graphs that are edge-transitive and have several classes of non-edges

Bandwidth of Johnson graphs ...

- Let Ω be a set of size v
- $\{$vertices of the Johnson graph $J(v,d)\} = \binom{\Omega}{d}$ and two subsets are adjacent if their intersections has size $d-1$

v	$\sharp$ nodes	$\mathrm{bw_L}$	$\mathrm{bw_{QAP}}$	time(s)	$\mathrm{bw_{new}}$	time(s)	u.b.
6	20	11	13	0	13	-	13
7	35	17	22	1	22	-	22
8	56	26	29	2	31	194	34
9	84	38	40	6	43	558	49
10	120	52	53	15	57	865	68

Table: Bounds on the bandwidth of $J(v,3)$

- lower bounds: $m_3 + \left\lceil -\frac{1}{2} + \sqrt{2\alpha + \frac{1}{4}} \right\rceil$
- upper bounds obtained by improving Cuthill-McKee heuristic

Bandwidth of Hamming graphs ...

- Hamming graph $H(d,q)$

q	$\sharp$ nodes	bw_L	bw_{QAP}	time(s)	bw_{new}	time(s)	u.b.
3	27	10	10	0	12	44	13
4	64	22	22	3	25	176	33
5	125	43	43	15	47	536	84

Table: Bounds on the bandwidth of $H(3,q)$

- lower bounds: $m_3 + \left\lceil -\frac{1}{2} + \sqrt{2\alpha + \frac{1}{4}} \right\rceil$

The end