Efficient exact algorithms for Graph Partitioning Problems

A. Ridha Mahjoub, Denis Cornaz, Fabio Furini, Mathieu Lacroix, Enrico Malaguti, Sébastien Martin
Gaspard Monge Program (PGMO)

Call for projects 2014:

Graph Partitioning (GP) Problems

- Design algorithms from polyhedral combinatorics and mathematical programming to tackle GP and its variants.
- The existing approaches are mainly heuristics.
- A big challenge is to have general and exact methods that efficiently solve large-sized GP problems.
Universities involved:

- **Université Paris-Dauphine**, MATHIS team of LAMSADÉ (CNRS – UMR 7243).  
  **MATHIS** – Mathematical Programming and Discrete Structures.

- **Université Paris 13**, AOC team of LIPN (CNRS – UMR 7030).  
  **AOC** – Algorithms and Combinatorial Optimisation.

- **Université de Lorraine**, DOP team of LCOMS.  
  **DOP** – Decision and OPtimization.
**General context**

*Graph Partitioning (GP)* is a fundamental family of problems with applications in many areas. Given a graph, GP asks for a partition of the vertex set into pairwise disjoint subsets with some additional properties.

**Vertex $k$-separator problems (VKS):** find a min-cost subset of vertices such that, after their removal, the graph partitions into $k$ pairwise disconnected subset of vertices.

- **Balanced VKS (BVKS):** the difference of cardinality between all pairs of subsets of vertices must be bounded by a given value.
- **Cardinality Constraint VKS (MCVKS):** the cardinality of subsets of vertices must be less or equal than a given value.
- **Steiner VKS (SVKS):** given a set of special vertices, each subset of vertices must contain exactly one of them.
Real-world Applications

Networks

- Clustering in Social and Biological Networks
- Risk Management of Energy and Social Networks
- Data Analysis
- VLSI Physical Design
- ...

Parallel computing

- Decomposition
- Scheduling in Multi-Processor Systems
- ...

Fabio Furini
Example I: Social Networks

- A social network is a platform to build social relations among people who share interests, activities or backgrounds.
- Find “disconnected” subsets of users of “similar size”.

Fabio Furini

Vertex k-Separator
Example II: parallel simulation of physical movements

The movement of a physical system (electric circuit, motor for example) can be modeled by a differential algebraic equation systems.

To solve such a system using a parallel algorithm, it is necessary to decompose the associated Jacobian matrix (minizing the interface variables).
Literature Review

S. Martin,
Analyse structurelle des systèmes algébro-différentiels conditionnels : complexité, modèles et polyèdres,

M. Didi Biha and M.J. Meurs,
An Exact Algorithm for Solving the Vertex Separator Problem,

E. Balas and C. de Souza,
The vertex separator problem: a polyhedral investigation,

C. de Souza and E. Balas,
The vertex separator problem: algorithms and computations,

G. Karypis and V. Kumar,
Multilevel k-way Partitioning Scheme for Irregular Graphs,
**Problem Definition**

---

**Balanced Vertex $k$-Separator (BVKS)**

- Given a graph $G = (V, E)$ and two integer values $k$ and $q$.
- Find the smallest subset of vertices $V_0 \subseteq V$ such that:
  1. $V \setminus V_0$ can be partitioned into $k$ subsets $V_1, \ldots, V_k$ that are pairwise disconnected;
  2. the difference of cardinality between all pairs of subsets $V_i, V_j$, $i \neq j \in \{1, \ldots, k\}$, is bounded by $q$.

---

The BVKS is strongly NP-hard in general.
### Notations

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V$</td>
<td>set of vertices</td>
</tr>
<tr>
<td>$E$</td>
<td>set of edges</td>
</tr>
<tr>
<td>$G = (V, E)$</td>
<td>$</td>
</tr>
<tr>
<td>$K$</td>
<td>set of integers ${1, ..., k}$</td>
</tr>
<tr>
<td>$\mathcal{N}(w)$</td>
<td>set of vertices neighborhood of $w$</td>
</tr>
<tr>
<td>$\mathcal{N}(W)$</td>
<td>set of vertices neighborhood of $W$</td>
</tr>
<tr>
<td>$\tilde{\mathcal{N}}(w)$</td>
<td>$\mathcal{N}(w) \cup w$</td>
</tr>
<tr>
<td>$\tilde{\mathcal{N}}(W)$</td>
<td>$\mathcal{N}(W) \cup W$</td>
</tr>
<tr>
<td>$\tilde{\mathcal{N}}'(w)$</td>
<td>set of vertices not in the neighborhood of $w$</td>
</tr>
</tbody>
</table>
**Mathematical Formulations – Part I**

**Binary Variables**

\[ x^i_v = \begin{cases} 
1 & \text{if } v \in V_i \\
0 & \text{otherwise.} 
\end{cases} \quad v \in V, i \in K \]
Mathematical Formulations – part I

Compact Formulation (CF)

\[
\text{max} \sum_{i \in K} \sum_{v \in V} x^i_v
\]

\[
\sum_{i \in K} x^i_v \leq 1 \quad v \in V, \quad (1)
\]

\[
x^i_u + x^j_v \leq 1 \quad i \neq j \in K, uv \in E, \quad (2)
\]

\[
\sum_{v \in V} (x^i_v - x^j_v) \leq q \quad i \neq j \in K, \quad (3)
\]

\[
x^i_v \geq 0 \quad i \in K, v \in V, \quad (4)
\]

\[
x^i_v \in \{0, 1\} \quad i \in K, \forall v \in V.
\]
Convex Hull

\[ P(G, k, q) = \text{conv}(\{x \in \{0, 1\}^{kn} | x \text{ satisfies (1) – (4)}\}). \]

Proposition

\( P(G, k, q) \) is full dimensional if \( q > 0 \).

Proposition

Inequalities (4), \( x^i_v \geq 0 \) define facets of \( P(G, k, q) \), for all \( i \in K, v \in V \).
Polyhedral properties II – Compact Formulation

**Convex Hull**

\[ P(G, k, q) = \text{conv}(\{x \in \{0, 1\}^{kn} | x \text{ satisfies (1) – (4)}\}). \]

**Proposition**

Inequality (2), \[ \sum_{i \in K} x_v^i \leq 1 \], associated with \( v \in V \), defines facets of \( P(G, k, q) \) if and only if:

- either \( q > 1 \)
- or for all edges \( uv \in E \), there exists in \( V \setminus \tilde{N}(\{u, v\}) \) an independent set of cardinality greater or equal to \( k - 1 \).
Proposition

Let $uv \in E$ be an edge and $i \in K$. The inequality

$$x_u^i + \sum_{j \in K \setminus \{i\}} x_v^j \leq 1,$$

is valid for $P(G, k, q)$.

Proposition

These inequalities define facets of $P(G, k, q)$ if and only if:

- either $q > 1$
- or for all edges $uv \in E$, there exists in $V \setminus \tilde{N}(\{u, v\})$ an independent set of cardinality greater or equal to $k - 1$. 
Valid Inequalities (A) – Compact Formulation

**W-balanced inequalities**

**Proposition**

Let \( W \subseteq V \) be the set of vertices such that for all \( v \in W \): either \(|\widetilde{N}'(v)| < q + k - 1\) or \(\widetilde{N}'(v)\) does not contain an independent set of cardinality \(k - 2\).

The following inequalities are valid for \( P(G, k, q) \):

\[
\sum_{v \in V} x^i_v - \sum_{u \in V \setminus W} x^j_u \leq q \quad i \neq j \in K, \tag{6}
\]

**Proposition**

These inequalities define facets of \( P(G, k, q) \) if and only if every vertex of \( W \) has a degree less than or equal to \( n - q - 1 \).
Valid Inequalities – Example

\[ q = 4, \ k = 2, \ W = \{w, v\} \]

\[ G = (V, E) \]

\[ |\overline{N}(w)| = 4 < 4 + 2 + 1 \]

\[ \text{ok} \]

\[ |\overline{N}(v)| = 3 < 4 + 2 - 1 \]

\[ \text{ok} \]

\[ \sum x^i_w - \sum_{u \in V} x^j_w \leq q \quad i \neq j \in K \]

\[ \sum_{u \in V} x^i_w - \sum_{u \in V \setminus w} x^j_w \leq q \quad i \neq j \in K \]
Proposition

Let $D_W$ be a connected dominating set of vertices of $G(W)$ ($W \subseteq V$) such that:

- $|D_W| > q$
- $|W| > n - (k - 1)|D_W| + (k - 1)q$.

The following inequalities are valid for $P(G, k, q)$:

$$
\sum_{v \in D_W} \sum_{i \in K} x^i_v \leq |D_W| - 1
$$

(7)
Valid Inequalities – Example

$G = (V, E)$

$q = 2 \quad k = 2 \quad D_w = \{m, n, 2\} \quad m = 8 \quad W = V$

$|D_w| = 3 > 2$

$|W| = 8 > 8 - 3 + 2$

$\sum \sum x_{v_i}^i \leq |D_w| - 1$

$x_w^2 + x_n^2 + x_2^2 + x_m^2 + x_v^2 + x_z^2 \leq 2$
Model (1)–(4)

A (trivial) fractional solution of value $|V|$, which is obtained by splitting vertices between 2 subsets so as to satisfy constraints (2).

Model (1)–(4) + (5)

A (trivial) fractional solution of the same value is obtained even if the model is strengthened; in this case each vertex is equally split among $k$ subsets.

Necessity of a stronger model!!!
Let $S = \{S \subseteq V\}$ be the family of all possible subsets of vertices of $V$.

**Binary Variables**

$$z^i_S = \begin{cases} 
1 & \text{if } S = V_i \\
0 & \text{otherwise.} 
\end{cases} \quad S \in S, \ i \in K$$
Mathematical Formulations – part II

Extended Formulation

\[
\text{max} \sum_{i \in K} \sum_{S \in S} |S| z^i_S \\
\sum_{i \in K} \sum_{S \in S: v \in S} z^i_S \leq 1 \quad v \in V, \\
\sum_{i \in K} \sum_{S \in S} a_{uv}^S z^i_S \leq 1 \quad uv \in E, \\
\sum_{S \in S} z^i_S = 1 \quad i \in K, \\
\sum_{S \in S} |S| z^i_S - \sum_{S \in S} |S| z^j_S \leq q \quad i \neq j \in K, \\
z^i_S \in \{0, 1\} \quad i \in K, S \in S.
\]
\[ \alpha_{uv}^S = \begin{cases} 
1 & \text{if } u \in S \lor v \in S, \\
0 & \text{otherwise}. 
\end{cases} \]
Mathematical Formulations – part II

Model (8)–(13) has an exponential number of variables, thus we need a column generation procedure to solve its continuous relaxation.

**Separation Problem**

Given the values of the dual variables $\lambda^*_v, \gamma^*_i, \pi^*_{uv}, \rho^*_{ij}$, associated with constraints (9), (10), (11) and (12), respectively.

For each $i \in K$, we look for a subset $S^* \in S$ such that:

$$\sum_{v \in S^*} \lambda^*_v + \sum_{uv \in E} a^*_{uv} \pi^*_{uv} + \gamma^*_i + |S^*| \sum_{j \in K: j \neq i} (\rho^*_{ij} - \rho^*_{ji}) < |S^*|$$

$$\sum_{v \in S^*} (\lambda^*_v + b^*_i - 1) + \sum_{(u,v) \in E} a^*_{uv} \pi^*_{uv} < -\gamma^*_i$$
The separation problem associated with $i \in K$ can be tackled as an optimization problem, denoted as SP in the following.

SP can be modeled as a Binary Linear Program using variables $x_v$, which determine whether vertex $v$ belongs to $S^*$, and variables $y_{uv}$, which model coefficient $a_{S^*}^{uv}$.

$$x_v = \begin{cases} 1 & v \in S^*, \\ 0 & \text{otherwise}, \end{cases} \quad v \in V, \quad y_{uv} = \begin{cases} 1 & u \in S^* \lor v \in S^*, \\ 0 & \text{otherwise}, \end{cases} \quad uv \in E.$$
Given $G = (V, E)$, a profit $\nu_v^*$ for each $v \in V$ and a penalty $\pi_{uv}^* \geq 0$ for each $(u, v) \in E$:

$$\max \sum_{v \in V} \nu_v^* x_v - \sum_{uv \in E} \pi_{uv}^* y_{uv}$$

(14)

$$y_{uv} \geq x_u \quad uv \in E,$$

(15)

$$y_{uv} \geq x_v \quad uv \in E,$$

(16)

$$x_v \in \{0, 1\} \quad v \in V,$$

(17)

$$y_{uv} \in \{0, 1\} \quad uv \in E,$$

(18)

where:

$$\nu_v^* = -\lambda_v^* - b_i^* + 1.$$
The SP problem aims at selecting the subset of vertices of maximum profit; if one vertex $v$ is selected, the penalty $\pi^*_{uv}$ associated with edges $uv \in E$ is paid and the profit $\nu^*_v$ is taken.
We may assume $\nu_v > 0$ for each $v \in V$. A vertex $v$ with $\nu_v \leq 0$ can be removed together with its incident edges (the profits $\nu_{vu}$ go to vertex $u$).

Since $\pi_{uv} \geq 0$, and $x_v \in \{0, 1\}$, we can replace (18) with $y \geq 0$.

**Proposition**

*The SP is polynomial-time solvable.*

Indeed, the transpose of the constraint matrix of model (14)–(18) has at most one coefficient of value 1 and one coefficient of value -1 per column. Thus, it is a *totally unimodular matrix* (TUM).

Since the transpose of a TUM is a TUM, the constraint matrix is TU and the linear relaxation of the SP has integer optimal solution.
Dual of SP I

\[
\begin{align*}
\min & \sum_{v \in V} \rho_v \\
\rho_v + \sum_{u : vu \in E} \gamma_{vu} &= \nu_v^* \quad v \in V \\
\gamma_{uv} + \gamma_{vu} &\leq \pi_{uv}^* \quad vu \in E \\
\rho_v &\geq 0 \quad v \in V \\
\gamma_{vu}, \gamma_{uv} &\geq 0 \quad vu \in E
\end{align*}
\]

\[
\begin{align*}
\max & \sum_{vu \in E} (\gamma_{vu} + \gamma_{uv}) + \phi \\
\sum_{u : vu \in E} \gamma_{vu} &\leq \nu_v^* \quad v \in V \\
\gamma_{uv} + \gamma_{vu} &\leq \pi_{uv}^* \quad vu \in E \\
\gamma_{vu}, \gamma_{uv} &\geq 0 \quad vu \in E
\end{align*}
\]

This problem be solved as a max-flow from origin \( s \) to destination \( t \) on network with three layers as follows:
Dual of SP II

- **First layer**: nodes $n_{vu}^1$ for each edge $vu \in E$. An arc of capacity $\pi_{vu}^*$ connects $\sigma$ to $n_{ij}^1$.
- **Second layer**: two nodes $n_{vu}^2$ and $n_{uv}^2$ for each node $n_{vu}^1$. Arcs of infinite capacity connect $n_{vu}^1$ to $n_{vu}^2$ and $n_{uv}^2$.
- **Third layer**: nodes $n_v^3$ for each vertex $i \in V$. An arc of infinite capacity connects each node $n_{vu}^2$ to $n_v^3$. An arc of capacity $\nu_v^*$ connects $\tau$ to $n_v^3$. 
Branching Scheme I

- The branching imposes that either a vertex $v$ belongs to a partition $i$, or it does not.
- If the solution is fractional it exist $i$ and $v$ such that:

$$0 < \sum_{s \in S: v \in s} x^i_s < 1.$$  \hspace{1cm} (19)

Then the branching generates two nodes by imposing either:

$$\sum_{s \in S: v \in s} x^i_s = 0,$$  \hspace{1cm} (20)

or

$$\sum_{s \in S: v \in s} x^i_s = 1.$$  \hspace{1cm} (21)
**Proposition**

*The branching scheme is complete.*

- The branching scheme does not affect the structure of the SP:
  - **first case**, vertex $v$ is no longer available for partition $i$,
  - **second case**, vertex $v$ must be included into partition $i$ and it is no longer available in partitions $j \in K, j \neq i$.

- The variables in Model (8)–(13) which are not consistent with the performed branching must then be removed.
## Experimental results I – Compact Formulation

<table>
<thead>
<tr>
<th>Instances</th>
<th>Basic model</th>
<th>Strengthen model</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$k$</td>
<td>$q$</td>
</tr>
<tr>
<td>myciel3</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>10</td>
</tr>
</tbody>
</table>
## Experimental results II – Compact Formulation

<table>
<thead>
<tr>
<th>Instances</th>
<th>Basic model</th>
<th>Strengthen model</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Name</td>
<td>$k$</td>
</tr>
<tr>
<td>myciel5</td>
<td>$</td>
<td>V</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10</td>
</tr>
</tbody>
</table>
## Experimental results III – Extended Formulation

<table>
<thead>
<tr>
<th>Instance</th>
<th>$k$</th>
<th>$q$</th>
<th>LP</th>
<th>gap</th>
<th>$Col_{LP}$</th>
<th>CPU</th>
<th>$Col_{B&amp;P}$</th>
<th>nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>myciel3</td>
<td>2</td>
<td>5</td>
<td>9</td>
<td>22,22</td>
<td>56</td>
<td>0</td>
<td>779</td>
<td>131</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>10</td>
<td>9,7</td>
<td>17,53</td>
<td>52</td>
<td>0</td>
<td>204</td>
<td>32</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>5</td>
<td>6,7</td>
<td>10,45</td>
<td>140</td>
<td>0</td>
<td>146</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>10</td>
<td>7,3</td>
<td>17,81</td>
<td>100</td>
<td>0</td>
<td>146</td>
<td>17</td>
</tr>
</tbody>
</table>
Conclusions

- We presented a compact Integer Linear Programming formulation and discuss some possible improvement which brought to a strengthen model. Then we introduce an exponential-size formulation, for which we derive a column generation and a branching scheme.

- Preliminary computational results comparing the performance of the two formulations on a set of benchmark instances are reported which give some insight about the difficulty of the problem and the dimension of the instance which can be solved to proven optimality.

- We especially evaluated the effect of the number of partitions on the CPU time.
Future works

- Further strengthening of the compact formulation
- New separation algorithms
- Testing alternative algorithmic solutions of the column generation problem (e.g., as a flow problem)
- Alternative branching schemes

Other Graph Partitioning Variants