Social Network Analysis and Community Detection by Decomposing a Graph into Relaxed Cliques

Column Generation 2016, Búzios, Brazil

Timo Gschwind, Stefan Irnich, Fabio Furini[1], Roberto Wolfler Calvo[2]

{gschwind,irnich}@uni-mainz.de

Chair for Logistics Management

Gutenberg School of Management and Economics

JOHANNES GUTENBERG UNIVERSITÄT MAINZ

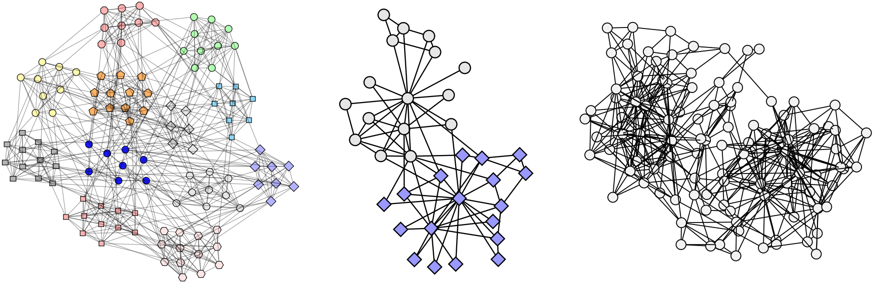
May 22 – 25, 2015

[1] fabio.furini@dauphine.fr, LAMSADE, Université Paris Dauphine

[2] roberto.wolfler@lipn.univ-paris13.fr, LIPN, Université Paris 13

Network Analysis:

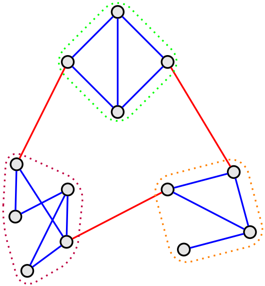
Graphs representing real systems are no random graphs
→ community structure or clustering



Important applications in many networked systems from biology, sociology, computer science, engineering, economics, politics, linguistics, etc.

Network Analysis:

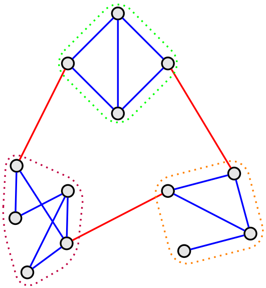
1. Community Detection (Fortunato, 2010)
   - Divide the graph into groups
   - Few edges between groups, relatively many internal edges
   - Maximize modularity (Newman and Girvan, 2004)
   - Groups have no specific structure



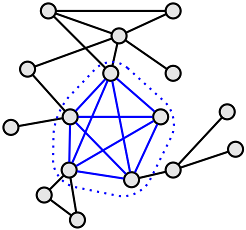
2. Relaxed Cliques (Pattillo *et al.*, 2013a)
   - Subgraphs with a specific structure
   - Find maximum cardinality/weight relaxed clique

Network Analysis:

1. Community Detection (Fortunato, 2010)
   - Divide the graph into groups
   - Few edges between groups, relatively many internal edges
   - Maximize modularity (Newman and Girvan, 2004)
   - Groups have no specific structure



2. Relaxed Cliques (Pattillo *et al.*, 2013a)
   - Subgraphs with a specific structure
   - Find maximum cardinality/weight relaxed clique



Outline

1. Introduction and problem description
2. Branch-and-Price algorithm
   - Master program and pricing problem
   - Branching
3. Computational results
4. Conclusions

Basic Notation

We assume that a simple graph $G = (V, E)$ with finite node set V and edge set E is given.

For any subset $S \subseteq V$, the subgraph $G[S] := (S, E \cap (S \times S))$ is called the induced subgraph of S.

A set S is a clique if $G[S]$ is complete.
A set S is an independent (stable) set if $G[S]$ is edgeless.
A clique/independent set is maximum if there is no clique/independent set with larger cardinality.

Examples:

Maximum Clique

Maximum Independent Set

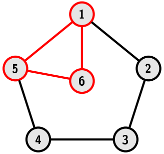
Basic Notation

We assume that a simple graph $G = (V, E)$ with finite node set V and edge set E is given.

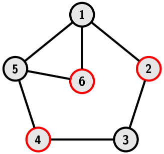
For any subset $S \subseteq V$, the subgraph $G[S] := (S, E \cap (S \times S))$ is called the induced subgraph of S.

A set S is a clique if $G[S]$ is complete.
A set S is an independent (stable) set if $G[S]$ is edgeless.
A clique/independent set is maximum if there is no clique/independent set with larger cardinality.

Examples:



Maximum Clique



Maximum Independent Set

Cliques and Clique Relaxations

A clique S forms an extreme subset in the following senses:

Degree	Every node $i \in S$ has maximum degree $(=|S|-1)$
Distance	The distance $d(i,j)$ between any two nodes $i,j \in S$ is minimal $(=1)$
Density	$G[S]$ has maximum density $(=1)$; Note: density of (V', E') is $\frac{|E'|}{\binom{|V'|}{2}}$
Connectivity	The vertex connectivity $\kappa(G[S])$ is maximum $(=|S|-1)$

Network analysis:

- Cliques can model cohesive substructures/communities (Luce and Perry, 1949).
- However, requirements of a clique were found too restrictive!

Cliques and Clique Relaxations

Different clique relaxations have been considered in the literature (see Pattillo *et al.*, 2013a, for an overview):

Type of relaxation	Definition	Based on	Hered-itary	Connected
k-core	$\delta(G[S]) \geq k$	Degree	no	$\lvert S\rvert \leq 2k+1$
s-plex	$\delta(G[S]) \geq \lvert S\rvert - s$	Degree	yes	$\lvert S\rvert \geq 2s-1$
s-clique	$d_G(i,j) \leq s \;\forall\; i,j \in S$	Dist.	yes	$s=1$
s-club	$d_{G[S]}(i,j) \leq s \;\forall\; i,j \in S$	Dist.	no	always
γ-quasi-clique	$\rho(G[S]) \geq \gamma$	Density	no	$\left\lceil \gamma\binom{\lvert S\rvert}{2} - \binom{\lvert S\rvert-1}{2} \right\rceil \geq 1$
s-defective cl.	$\lvert E(G[S])\rvert \geq \binom{\lvert S\rvert}{2} - s$	Density	yes	$\lvert S\rvert \geq s+2$
k-block	$\kappa(G[S]) \geq k$	Connect.	no	always
s-bundle	$\kappa(G[S]) \geq \lvert S\rvert - s$	Connect.	yes	$\lvert S\rvert \geq s+1$

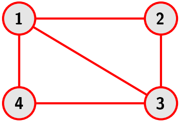
Note: $\delta(G)$ minimum degree, $d_G(i,j)$ distance, $\rho(G)$ edge density, $\kappa(G)$ vertex connectivity

Denote by Π the graph property of being a specific type of relaxed clique.

Clique Relaxations: Examples

1-defective clique

'one missing edge allowed in induced subgraph'



$\Rightarrow$ **2-plex**

'one missing edge allowed at each node'

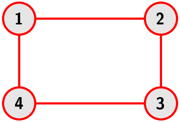
2-club

'distance between nodes in S at most 2 in induced subgraph'

$\Rightarrow$ **2-clique**

'distance between nodes of S at most 2 in graph'

2-plex



$\not\Rightarrow$ 1-defective clique

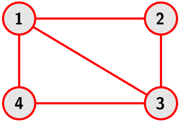
2-clique

$\not\Rightarrow$ 2-club

Clique Relaxations: Examples

1-defective clique

'one missing edge allowed in induced subgraph'

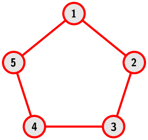


$\Rightarrow$ **2-plex**

'one missing edge allowed at each node'

2-club

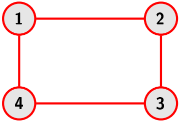
'distance between nodes in S at most 2 in induced subgraph'



$\Rightarrow$ **2-clique**

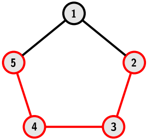
'distance between nodes of S at most 2 in graph'

2-plex



$\not\Rightarrow$ 1-defective clique

2-clique



$\not\Rightarrow$ 2-club

Weak-Heredity

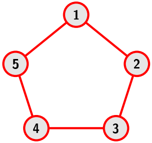
S with satisfies Π
$\Rightarrow$ any $S' \subset S$ satisfies Π.

Weak-Hereditary relaxed cliques:

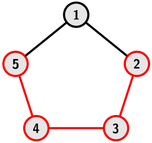
s-plex, s-clique, s-defective clique, s-bundle

Non-hereditary relaxed cliques:

k-core, s-club, γ-quasi-clique, k-block



2-club



no 2-club

Connectivity

For all $i, j \in S$ there exists a path between i and j in $G[S]$.

Connected relaxed cliques:

s-club, k-block

Non-connected relaxed cliques:

k-core, s-plex, s-clique, s-bundle, γ-quasi-clique, s-defective clique

2-defective clique

Note:

- Non-connected communities may not be reasonable
  $\rightarrow$ **Connected relaxed cliques**
- Connectivity is non-hereditary

Weak-Heredity

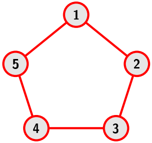
S with satisfies Π
$\Rightarrow$ any $S' \subset S$ satisfies Π.

Weak-Hereditary relaxed cliques:

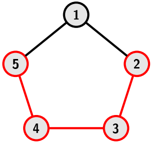
s-plex, s-clique, s-defective clique, s-bundle

Non-hereditary relaxed cliques:

k-core, s-club, γ-quasi-clique, k-block



2-club



no 2-club

Connectivity

For all $i, j \in S$ there exists a path between i and j in $G[S]$.

Connected relaxed cliques:

s-club, k-block

Non-connected relaxed cliques:

k-core, s-plex, s-clique, s-bundle, γ-quasi-clique, s-defective clique

2-defective clique

Note:

- Non-connected communities may not be reasonable
  $\rightarrow$ **Connected relaxed cliques**
- Connectivity is non-hereditary

Large Relaxed Cliques

The optimization-related literature related to relaxed cliques is (as far as we know) exclusively on finding maximum/inclusion maximal relaxed cliques.

Type of relaxation	Exact Approach and Reference
k-core	polynom. solvable, see (Kosub, 2004)
s-plex	B&C: (Balasundaram *et al.*, 2011), B&B: (Trukhanov *et al.*, 2013; Gschwind *et al.*, 2015)
s-clique	clique in the sth power graph
s-club	B&C: (Almeida and Carvalho, 2012, 2013), B&B: (Bourjolly *et al.*, 2002; Mahdavi Pajouh and Balasundaram, 2012), MIP: (Bourjolly *et al.*, 2000; Veremyev and Boginski, 2012), SAT: (Wotzlaw, 2014)
γ-quasi-clique	MIP: (Pattillo *et al.*, 2013b), B&B: (Pajouh *et al.*, 2014)
s-defective clique	B&C: (Sherali and Smith, 2006), B&B: (Trukhanov *et al.*, 2013; Gschwind *et al.*, 2015)
k-block	polynom. solvable, see (Kammer and Täubig, 2005)
s-bundle	B&B: (Gschwind *et al.*, 2015)

Large Relaxed Cliques

The optimization-related literature related to relaxed cliques is (as far as we know) exclusively on finding maximum/inclusion maximal relaxed cliques.

Generic MIP-formulation:

- $x_i \in \{0, 1\}$ indicates if vertex $i \in V$ is in the relaxed clique S
- $y_e \in \{0, 1\}$ indicates if $G[S]$ contains edge $e \in E$

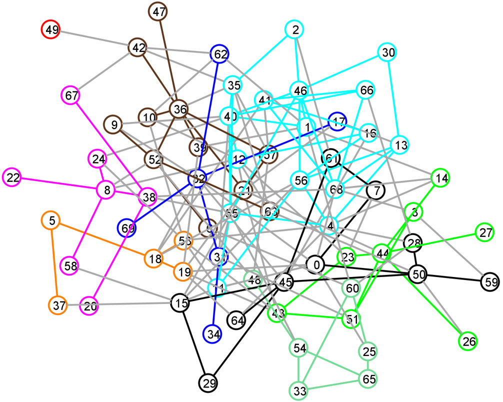
$$\max \quad \sum_{i \in V} x_i \tag{1}$$

$$\text{s.t.} \quad (x, y) = (x_i, y_{ij}) \in \mathscr{F}(G) \tag{2}$$

Note: $\mathscr{F}(G)$ is a polytope such that $(x, y) \in \mathscr{F}(G)$ if and only if $G[S]$ with $S = \{i \in V : x : i = 1\}$ satisfies Π.

The problems we consider in the following are partitioning and covering a graph with a minimum number of relaxed cliques:

Example: Partitioning with 3-Clubs



Special case:

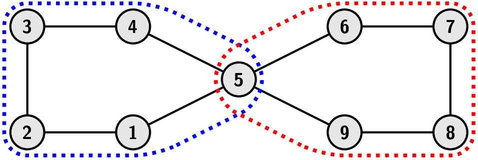
1-club = $(|S|-1)$-core = 1-plex = $\cdots$ = clique

- Clique Cover
- Vertex Coloring on Complement Graph

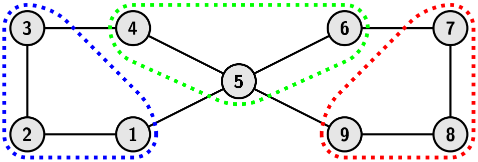
The problems we consider in the following are partitioning and covering a graph with a minimum number of relaxed cliques:

Partitioning $\neq$ **covering** for non-hereditary Π

Covering Solution:
Two 2-Clubs



Partitioning Solution:
Three 2-Clubs



Partitioning and Covering a Graph with Relaxed Cliques

The problems we consider in the following are partitioning and covering a graph with a minimum number of relaxed cliques:

Generic **compact formulation**:

- Let UB_{RC} be an upper bound for the problem
- Index set $H = \{1, ..., UB_{RC}\}$ to refer to the individual relaxed cliques in the partitioning/covering
- Introduce indicator variables $z^h \in \{0, 1\}$, $h \in H$
- Duplicate x_i and y_e variables and constraints for each $h \in H$

$$
\begin{aligned}
\min \quad & \sum_{h \in H} z^h \\
\text{s.t.} \quad & \sum_{h \in H} x_i^h = (\geq) 1 && i \in V \\
& z^h \geq x_i^h && i \in V, h \in H \\
& (x_i^h, y_{ij}^h) \in \mathscr{F}(G) && h \in H
\end{aligned}
$$

Interesting problem variants:

	General		With connectivity required	
Type of relaxation	Partitioning	Covering	Partitioning	Covering
k-core	✗	✗	✗	✗
s-plex	✔	✗	✔	✔
s-clique	✗	✗	✔	✔
s-club	✔	✔	✗	✗
γ-quasi-clique	✔	✔	✔	✔
s-defective clique	✔	✗	✔	✔
k-block	✗	✗	✗	✗
s-bundle	✔	✗	✔	✔

Note:

- ✗ (blue) Some vertices may have a degree smaller than k
- ✗ (green) Determination of k-connected components (Kammer and Täubig, 2005)
- ✗ (orange) Clique cover in sth power graph
- ✗ (red) Partitioning = covering for hereditary Π
- ✗ (purple) s-Club is always connected

Master problem derived from Dantzig-Wolfe decomposition of the generic compact formulation and subsequent aggregation:

- Ω the set of all S satisfying Π
- Indicators a_{iS} for $i \in V, S \in \Omega$ with $a_{iS} = 1$ if $i \in S$, and 0 otherwise

$$
\begin{aligned}
\min \quad & \sum_{S \in \Omega} \mathbb{1}^\top \lambda_S \\
\text{s.t.} \quad & \sum_{S \in \Omega} a_{iS} \lambda_S = 1 \quad (\text{or} \geq 1) \quad \text{for all } i \in V \qquad (\text{dual } \pi_i) \\
& \lambda \geq \mathbf{0} \quad (\in \mathbb{R}^{|\Omega|})
\end{aligned}
$$

Pricing Problem

Reduced Costs (rdc) of a relaxed clique S:

$$\tilde{c}_S = 1 - \sum_{i \in S} \pi_i$$

Pricing Problem:

$$1 - \tilde{c}_S = \max \quad \sum_{i \in V} \pi_i x_i \qquad \text{s.t.} \quad (x_i, y_{ij}) \in \mathscr{F}(G)$$

This is a maximum weight relaxed clique problem (MW-RC)

- Generalization of the maximum (cardinality) relaxed clique problem
- For most types of relaxed cliques this has not been studied in the literature so far
- MIP-solver (+ cutting plane algorithm) or combinatorial B&B
- Weights $\pi_i \in \mathbb{R}$ can be negative in case of partitioning

Reduced Costs (rdc) of a relaxed clique S:

$$\tilde{c}_S = 1 - \sum_{i \in S} \pi_i$$

Pricing Problem:

$$\begin{aligned} 1 - \tilde{c}_S = \max \quad & \sum_{i \in V} \pi_i x_i \\ \text{s.t.} \quad & (x_i, y_{ij}) \in \mathscr{F}(G) \end{aligned}$$

This is a maximum weight relaxed clique problem (MW-RC)

- Generalization of the maximum (cardinality) relaxed clique problem
- For most types of relaxed cliques this has not been studied in the literature so far
- MIP-solver (+ cutting plane algorithm) or combinatorial B&B
- Weights $\pi_i \in \mathbb{R}$ can be negative in case of partitioning

Branching for Relaxed Clique Partitioning

Branching schemes: (Pricing problem preserving)

- **Generic Branching Rule (GBR)**
  Branching on infeasible components of the support graph
  - Pros
    1. Applicable to all relaxed cliques
    2. Basic idea: easy to understand
    3. Simple to implement (remove edges from given graph)
  - Cons
    1. Complete only for connected weak-hereditary Π
    2. Non-binary scheme
- **Infeasible $(s+1)$-path branching**
  GBR tailored to s-club
  - Pros
    1. Has never failed in our experiments
    2. Simple to implement (remove edges from given graph)
  - Cons
    1. Non-binary scheme
- **Open:** Complete preserving branching scheme for other Π

Branching for Relaxed Clique Partitioning

Branching schemes: (Pricing problem preserving)

- **Generic Branching Rule (GBR)**
  Branching on infeasible components of the support graph
  - Pros
    1. Applicable to all relaxed cliques
    2. Basic idea: easy to understand
    3. Simple to implement (remove edges from given graph)
  - Cons
    1. Complete only for connected weak-hereditary Π
    2. Non-binary scheme

- **Infeasible** $(s+1)$**-path branching**
  GBR tailored to s-club
  - Pros
    1. Has never failed in our experiments
    2. Simple to implement (remove edges from given graph)
  - Cons
    1. Non-binary scheme

- **Open:** Complete preserving branching scheme for other Π

Branching for Relaxed Clique Partitioning

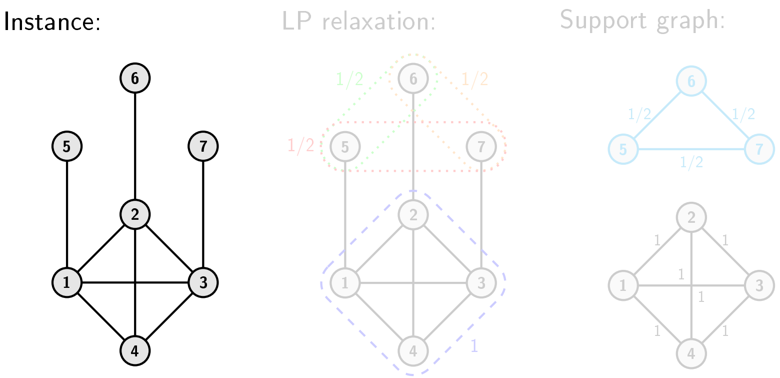
Branching schemes: (Pricing problem preserving)

- **Generic Branching Rule (GBR)**
  Branching on infeasible components of the support graph
  - Pros
    1. Applicable to all relaxed cliques
    2. Basic idea: easy to understand
    3. Simple to implement (remove edges from given graph)
  - Cons
    1. Complete only for connected weak-hereditary Π
    2. Non-binary scheme

- **Infeasible** $(s+1)$**-path branching**
  GBR tailored to s-club
  - Pros
    1. Has never failed in our experiments
    2. Simple to implement (remove edges from given graph)
  - Cons
    1. Non-binary scheme

- **Open:** Complete preserving branching scheme for other Π

Example 1: Partitioning with 2-plex (non-connected):



Non-integer part of the solution is disconnected in the original graph

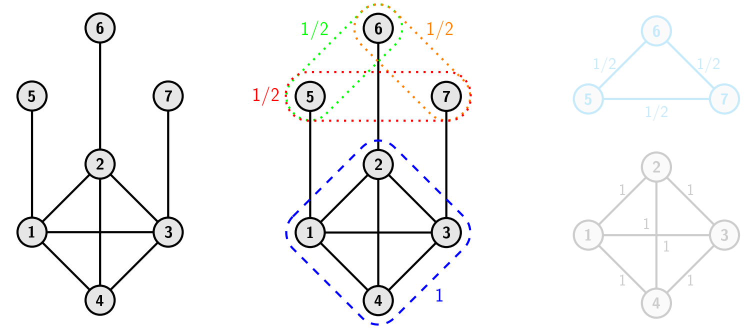
⇒ All components of the support graph are 2-plex

⇒ **GBR fails**

Branching for Relaxed Clique Partitioning

Example 1: Partitioning with 2-plex (non-connected):

Instance: LP relaxation: Support graph:



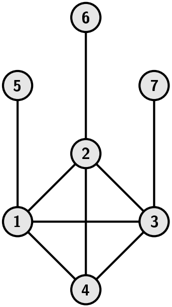
Non-integer part of the solution is disconnected in the original graph

⇒ All components of the support graph are 2-plex

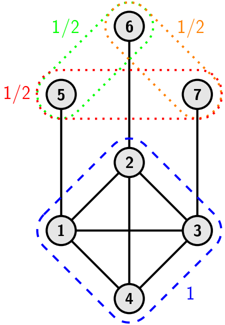
⇒ **GBR fails**

Example 1: Partitioning with 2-plex (non-connected):

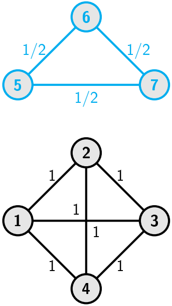
Instance:



LP relaxation:



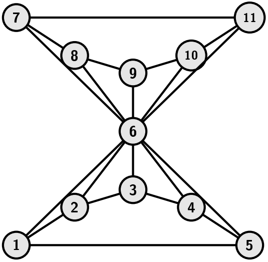
Support graph:



Non-integer part of the solution is disconnected in the original graph
$\Rightarrow$ All components of the support graph are 2-plex
$\Rightarrow$ **GBR fails**

Example 2: Partitioning with 2/3-quasi-clique (non-hereditary)

Instance:



LP relaxation:

Support graph:

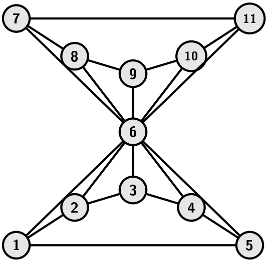
All connected components not fulfilling Π have a superset that satisfies Π.

⇒ **GBR fails**

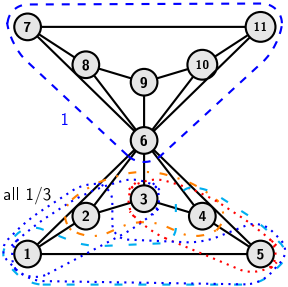
Branching for Relaxed Clique Partitioning

Example 2: Partitioning with 2/3-quasi-clique (non-hereditary)

Instance:



LP relaxation:



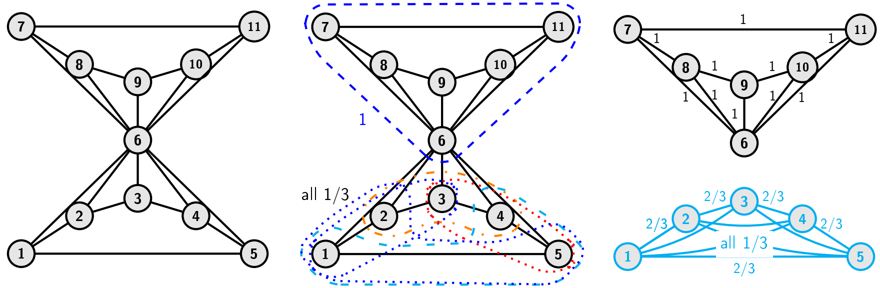
Support graph:

All connected components not fulfilling Π have a superset that satisfies Π.

⇒ **GBR fails**

Example 2: Partitioning with 2/3-quasi-clique (non-hereditary)

Instance: LP relaxation: Support graph:



All connected components not fulfilling Π have a superset that satisfies Π.

⇒ **GBR fails**

Branching for Relaxed Clique Partitioning

Branching schemes: (Pricing problem non-preserving)

Ryan and Foster (1981) branching
Branching on whether two nodes $i, j \in V$ are in the same ($x_i = x_j$) or different ($x_i + x_j \leq 1$) relaxed cliques

Pros
1. Complete
2. Easy to understand
3. Effective for set partitioning in general
4. Simple to handle in MIP-based pricing algorithms

Cons
1. Destroys structure of MW-RC pricing problem
2. Substantially complicates combinatorial B&B-based pricing algorithms

Branching for Relaxed Clique Partitioning

Branching schemes: (Pricing problem non-preserving)

Ryan and Foster (1981) branching
Branching on whether two nodes $i, j \in V$ are in the same ($x_i = x_j$) or different ($x_i + x_j \leq 1$) relaxed cliques

Pros
1. Complete
2. Easy to understand
3. Effective for set partitioning in general
4. Simple to handle in MIP-based pricing algorithms

Cons
1. Destroys structure of MW-RC pricing problem
2. Substantially complicates combinatorial B&B-based pricing algorithms

Branching for Relaxed Clique Covering

Let $g(T) = \sum_{S \in \Omega} |T \cap S| \lambda_S$ be the number of vertex contacts for subset $T \subseteq V$ and $g^\lambda(T)$ the value in the current solution.

Hierarchy of Branching Decisions:

1. Branch on vertex contacts for $T \subseteq V$ with branches $g(T) \leq \lfloor g^\lambda(T) \rfloor$ and $g(T) \geq \lceil g^\lambda(T) \rceil$
2. Fix vertex contacts of $P \subseteq V$ to its minimum ($g(P) = |P|$), alternative branch $g(P) \geq |P| + 1$
3. Apply branching for partitioning in $V_{=1}\{i \in V : g_i \text{ is fixed to } 1\}$
4. Vertex duplication for nodes in $V_{>1}\{i \in V : g_i \text{ is fixed to value } > 1\}$
   - Replace node i by g_i^λ copies $i_{(1)}, \ldots, i_{g_i^\lambda}$ of that node
   - Each of these node must be covered exactly once
   - No relaxed clique must contain two (or more) of these nodes
   - Separate constraints $x_{i_{(s)}} + x_{i_{(t)}} \leq 1$ for all pairs of copies $s \neq t$

Branching for Relaxed Clique Covering

Let $g(T) = \sum_{S \in \Omega} |T \cap S| \lambda_S$ be the number of vertex contacts for subset $T \subseteq V$ and $g^\lambda(T)$ the value in the current solution.

Hierarchy of Branching Decisions:

1. Branch on vertex contacts for $T \subseteq V$ with branches $g(T) \leq \lfloor g^\lambda(T) \rfloor$ and $g(T) \geq \lceil g^\lambda(T) \rceil$
2. Fix vertex contacts of $P \subseteq V$ to its minimum ($g(P) = |P|$), alternative branch $g(P) \geq |P| + 1$
3. Apply branching for partitioning in $V_{=1}\{i \in V : g_i \text{ is fixed to } 1\}$
4. Vertex duplication for nodes in $V_{>1}\{i \in V : g_i \text{ is fixed to value} > 1\}$
   - Replace node i by g_i^λ copies $i_{(1)}, \ldots, i_{g_i^\lambda}$ of that node
   - Each of these node must be covered exactly once
   - No relaxed clique must contain two (or more) of these nodes
   - Separate constraints $x_{i_{(s)}} + x_{i_{(t)}} \leq 1$ for all pairs of copies $s \neq t$

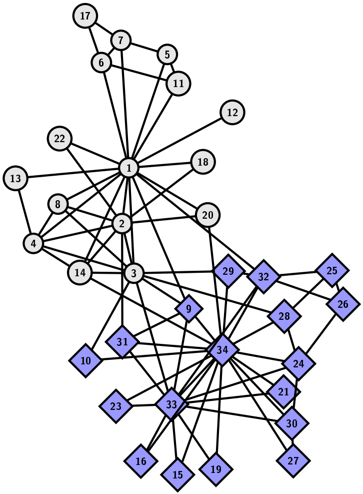
Branching for Relaxed Clique Covering

Let $g(T) = \sum_{S \in \Omega} |T \cap S| \lambda_S$ be the number of vertex contacts for subset $T \subseteq V$ and $g^\lambda(T)$ the value in the current solution.

Hierarchy of Branching Decisions:

1. Branch on vertex contacts for $T \subseteq V$ with branches $g(T) \leq \lfloor g^\lambda(T) \rfloor$ and $g(T) \geq \lceil g^\lambda(T) \rceil$
2. Fix vertex contacts of $P \subseteq V$ to its minimum ($g(P) = |P|$), alternative branch $g(P) \geq |P| + 1$
3. Apply branching for partitioning in $V_{=1}\{i \in V : g_i \text{ is fixed to } 1\}$
4. Vertex duplication for nodes in $V_{>1}\{i \in V : g_i \text{ is fixed to value } > 1\}$
   - Replace node i by g_i^λ copies $i_{(1)}, \dots, i_{g_i^\lambda}$ of that node
   - Each of these node must be covered exactly once
   - No relaxed clique must contain two (or more) of these nodes
   - Separate constraints $x_{i_{(s)}} + x_{i_{(t)}} \leq 1$ for all pairs of copies $s \neq t$

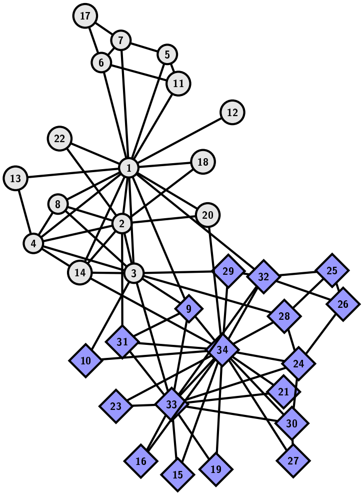
Zachary's Karate Club (Zachary (1977), also: 10th DIMACS challenge)



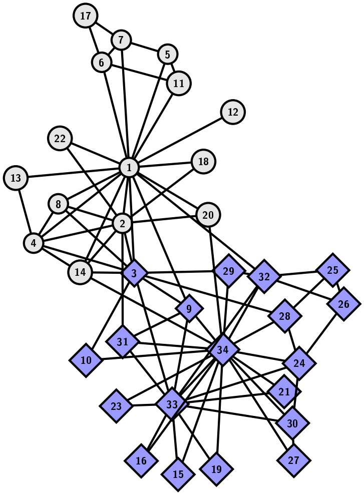
'Real' solution (Zachary, 1977)

3-club partitioning

Zachary's Karate Club (Zachary (1977), also: 10th DIMACS challenge)

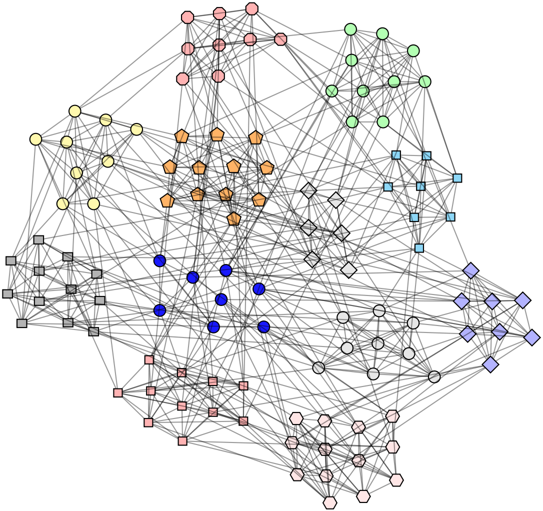


'Real' solution (Zachary, 1977)



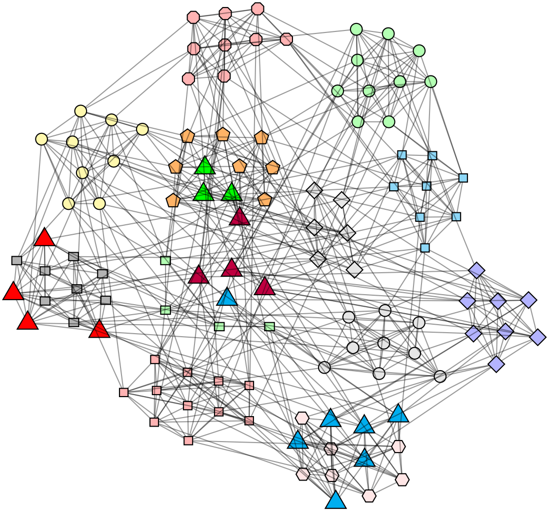
3-club partitioning

College Football (Girvan and Newman (2002), 10th DIMACS challenge)



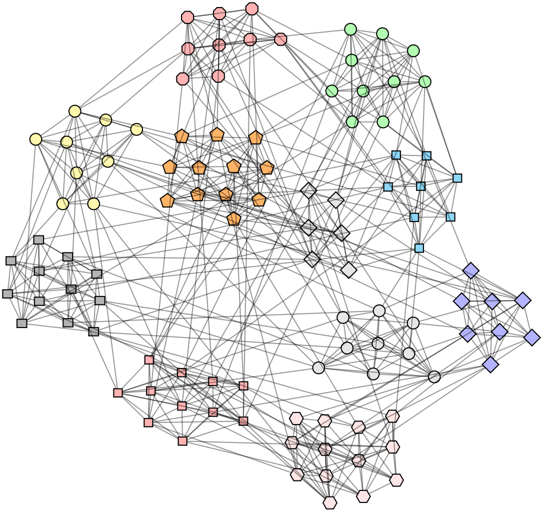
'Real' solution: 11 Conferences + 8 independent teams

College Football (Girvan and Newman (2002), 10th DIMACS challenge)



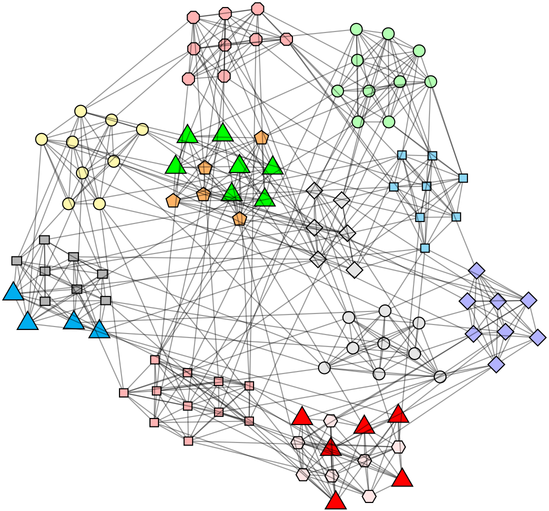
3-plex partitioning in 16 groups

College Football (no independents teams)



'Real' solution: 11 Conferences

College Football (no independents teams)



3-plex partitioning in 14 groups

Branch-and-Price insights:

1. Excellent lower bounds
2. Practical hardness increases with s (decreases with γ) and the density of the graph and depends on the type of relaxed clique
3. Covering vs. partitioning
   - Covering slightly easier than partitioning for solving LP-relaxation of the master program
   - Covering much harder when it comes to branching
4. Branching
   - Ryan/Foster branching seems to be much more effective
   - Pricing problems get harder when using Ryan/Foster branching
5. Subproblem solution
   - Combinatorial B&Bs much faster than MIPs for solving the pricing problem
   - Ryan/Foster branching complicates the pricing problems for the combinatorial B&Bs

Conclusions and Outlook

Conclusions:

- Introduced the relaxed clique covering/partitioning problem
- New approach for community detection
- Interesting components of branch-and-price
  - Branching
  - Solution of maximum weight relaxed clique pricing problem

Outlook:

- Finding a complete pricing problem preserving branching scheme
- Heuristics and metaheuristics can accelerate pricing

Conclusions and Outlook

Conclusions:

- Introduced the relaxed clique covering/partitioning problem
- New approach for community detection
- Interesting components of branch-and-price
  - Branching
  - Solution of maximum weight relaxed clique pricing problem

Outlook:

- Finding a complete pricing problem preserving branching scheme
- Heuristics and metaheuristics can accelerate pricing

Thank you for listening!

Any questions?!

Almeida, M. T. and Carvalho, F. D. (2012). Integer models and upper bounds for the 3-club problem. *Networks*, **60**(3), 155–166.

Almeida, M. T. and Carvalho, F. D. (2013). An analytical comparison of the LP relaxations of integer models for the k-club problem. *European Journal of Operational Research*, (0).

Balasundaram, B., Butenko, S., and Hicks, I. V. (2011). Clique relaxations in social network analysis: The maximum *k*-plex problem. *Operations Research*, **59**(1), 133–142.

Bourjolly, J.-M., Laporte, G., and Pesant, G. (2000). Heuristics for finding k-clubs in an undirected graph. *Computers & Operations Research*, **27**(6), 559–569.

Bourjolly, J.-M., Laporte, G., and Pesant, G. (2002). An exact algorithm for the maximum k-club problem in an undirected graph. *European Journal of Operational Research*, **138**(1), 21–28.

Brandes, U. and Erlebach, T., editors (2005). *Network Analysis: Methodological Foundations [outcome of a Dagstuhl seminar, 13-16 April 2004]*, volume 3418 of *Lecture Notes in Computer Science*. Springer.

Fortunato, S. (2010). Community detection in graphs. *Physics Reports*, **486**(3–5), 75–174.

Girvan, M. and Newman, M. E. J. (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, **99**(12), 7821–7826.

Gschwind, T., Irnich, S., and Podlinski, I. (2015). Maximum weight relaxed cliques and russian doll search revisited. Technical Report LM-2015-02, Chair of Logistics Management, Gutenberg School of Management and Economics, Johannes Gutenberg University Mainz, Mainz, Germany.

Kammer, F. and Täubig, H. (2005). Connectivity. In Brandes and Erlebach (2005), pages 143–177.

Kosub, S. (2004). Local density. In Brandes and Erlebach (2005), pages 112–142.

Luce, R. D. and Perry, A. D. (1949). A method of matrix analysis of group structure. *Psychometrika*, **14**(2), 95–116.

Mahdavi Pajouh, F. and Balasundaram, B. (2012). On inclusionwise maximal and maximum cardinality -clubs in graphs. *Discrete Optimization*, **9**(2), 84–97.

Newman, M. E. J. and Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical Review E*, **69**, 026113.

Pajouh, F. M., Miao, Z., and Balasundaram, B. (2014). A branch-and-bound approach for maximum quasi-cliques. **216**(1), 145–161.

Pattillo, J., Youssef, N., and Butenko, S. (2013a). On clique relaxation models in network analysis. *European Journal of Operational Research*, **226**(1), 9–18.

Pattillo, J., Veremyev, A., Butenko, S., and Boginski, V. (2013b). On the maximum quasi-clique problem. *Discrete Applied Mathematics*, **161**(1–2), 244–257.

Ryan, D. and Foster, B. (1981). An integer programming approach to scheduling. In A. Wren, editor, *Computer Scheduling of Public Transport: Urban Passenger Vehicle and Crew Scheduling*, chapter 17, pages 269–280. Elsevier, North-Holland.

Sherali, H. D. and Smith, J. C. (2006). A polyhedral study of the generalized vertex packing problem. *Mathematical Programming*, **107**(3), 367–390.

Trukhanov, S., Balasubramaniam, C., Balasundaram, B., and Butenko, S. (2013). Algorithms for detecting optimal hereditary structures in graphs, with application to clique relaxations. *Computational Optimization and Applications*, **56**(1), 113–130.

Veremyev, A. and Boginski, V. (2012). Identifying large robust network clusters via new compact formulations of maximum k-club problems. *European Journal of Operational Research*, **218**(2), 316–326.

Wotzlaw, A. (2014). On solving the maximum *k*-club problem. Technical Report arXiv:1403.5111v2, Institut für Informatik, Universität zu Köln, Köln, Germany.

Zachary, W. W. (1977). An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, **33**(4), 452–473.