Network-based Data Analysis

Sergiy Butenko
(butenko@tamu.edu)

Industrial and Systems Engineering, Texas A&M University

ACDL-2018, Siena, Italy, July 21-22, 2018

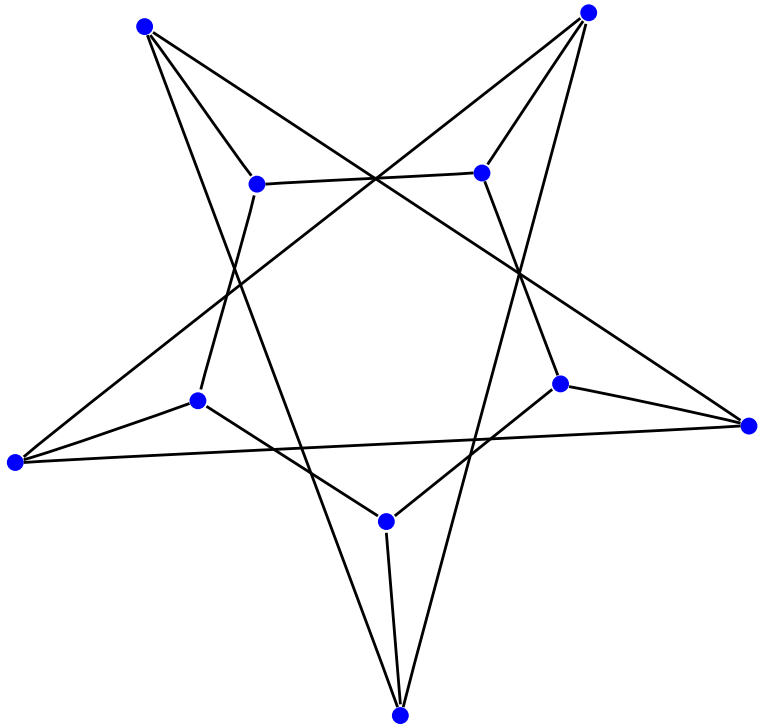
Outline

1. Network-based models of data
   - Introduction
   - Examples
   - Clusters in networks
2. Cluster-detection methods in network-based data analysis
   - Clique relaxations taxonomy
   - Optimization problems
3. Continuous approaches to cluster-detection problems
   - Continuous formulations of discrete problems
   - Complexity implications
   - Algorithmic issues
   - Future research directions

Outline

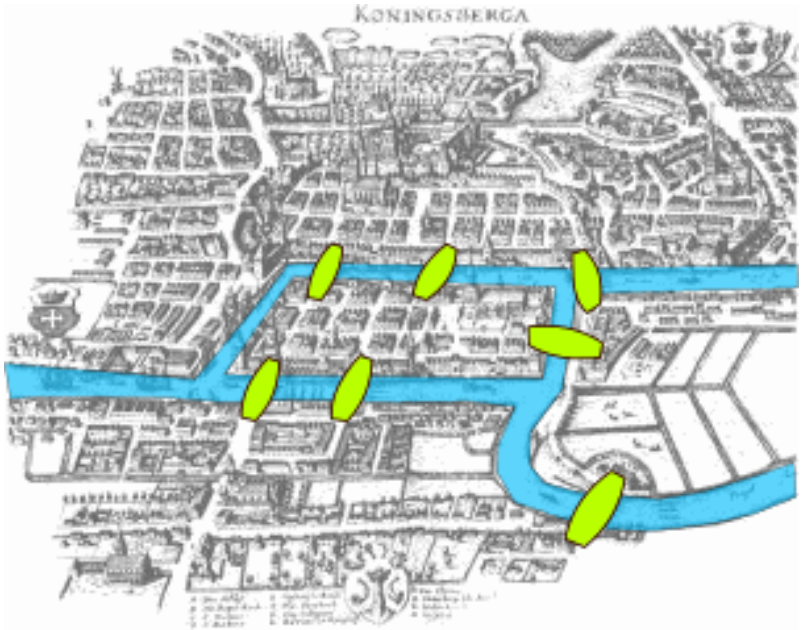
Graphs/networks

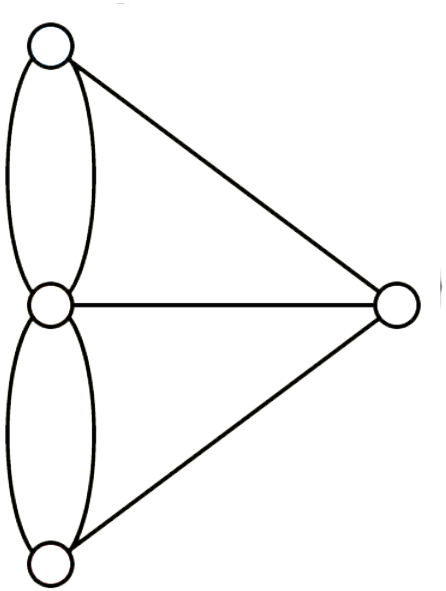
Graph is the mathematical term for a "network" - often visualized as vertices (points, nodes) connected by edges (lines, arcs)



Histrory

The origins of graph theory are attributed to the Seven Bridges of Königsberg problem solved by Leonhard Euler in 1735.





Network-based analysis of big data

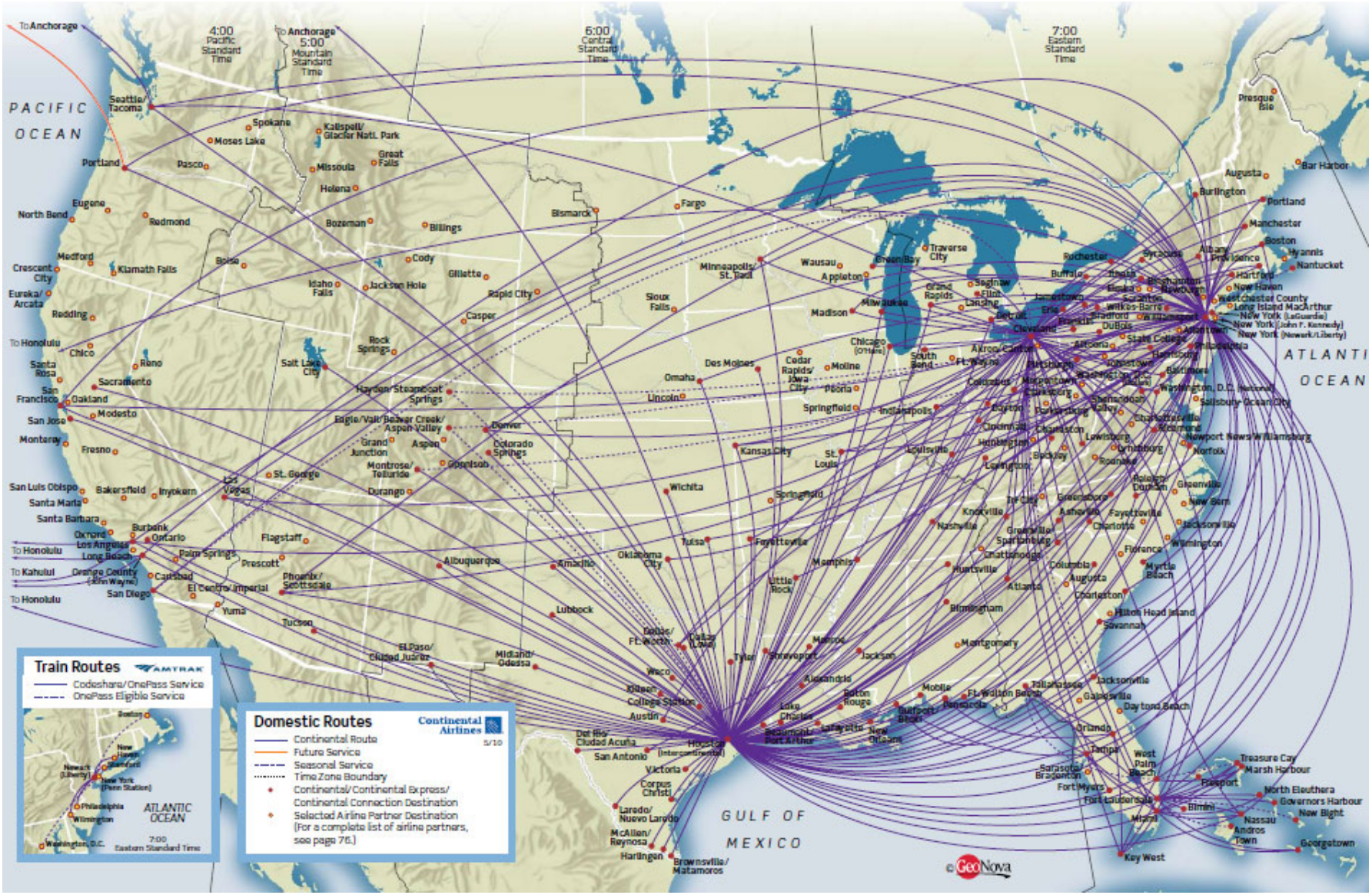
Big data arising in various complex systems can be conveniently modeled using networks/graphs:

- components of the complex system – vertices
- pairwise interactions between different components – edges

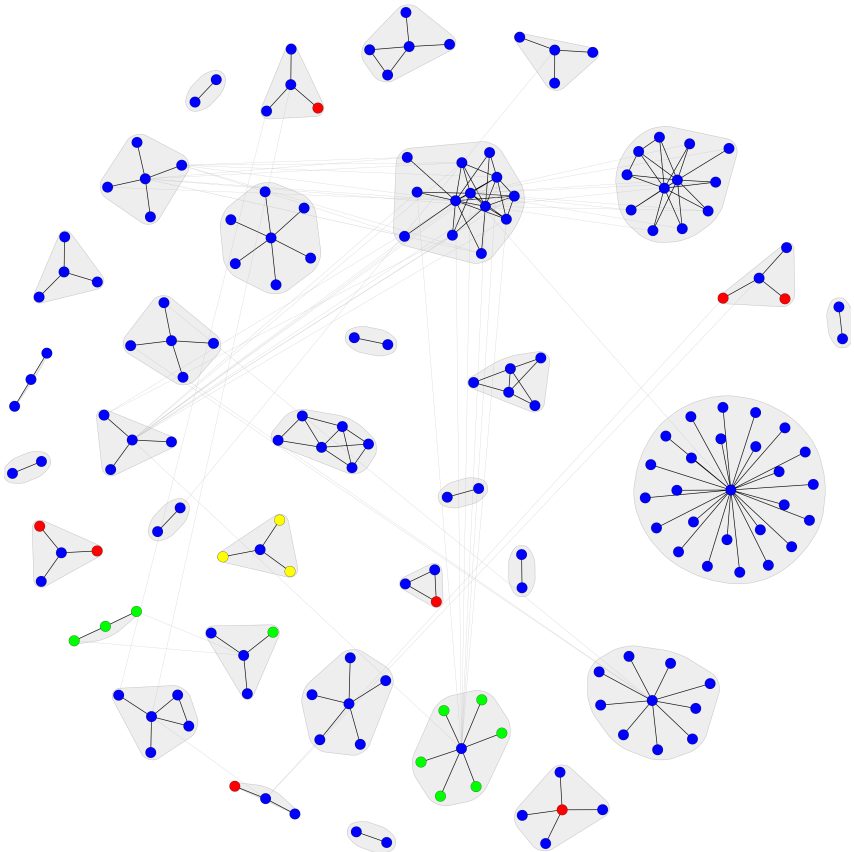
Network-based analysis allows to capture some global structural properties of the system and predict overall trends in its dynamics.

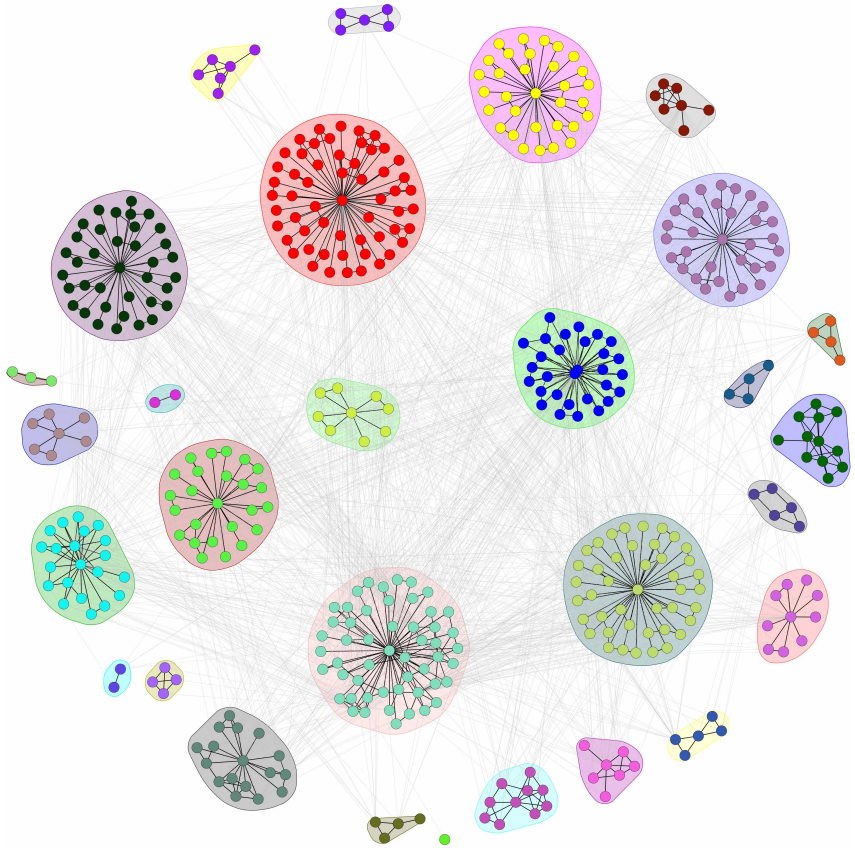
Outline

Transportation networks



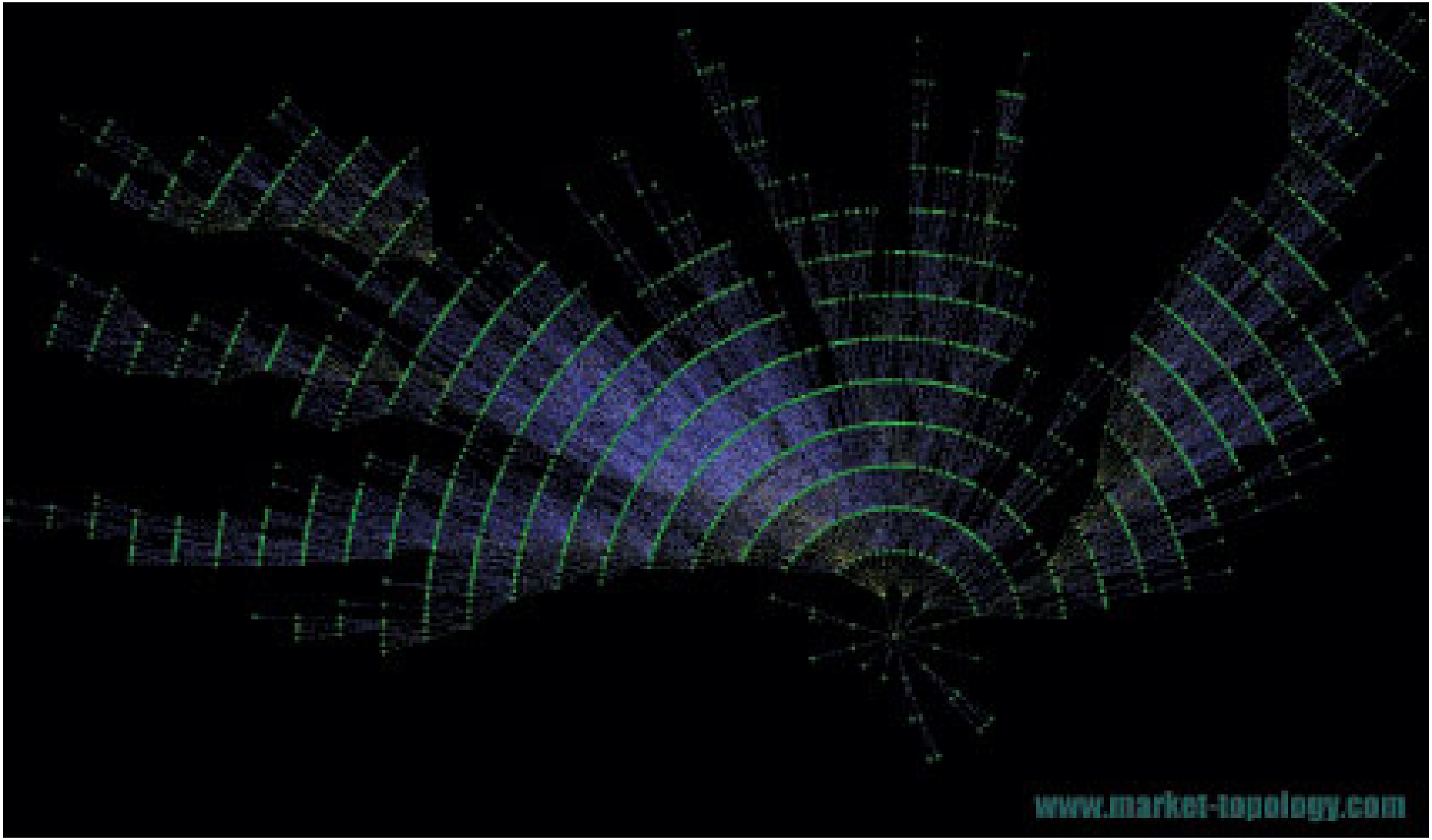
Biologicial networks



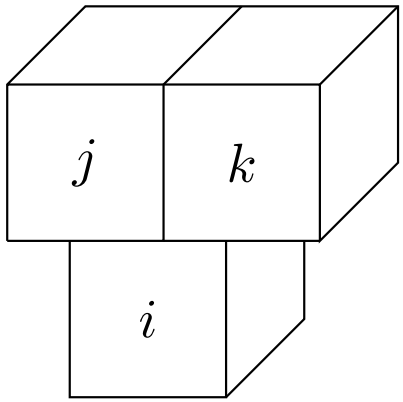


O. Yezerska, F. Mahdavi Pajouh, A. Veremyev, S. Butenko. Exact algorithms for the minimum s-club partitioning problem. *Annals of Operations Research*, to appear.

Financial networks



Open pit mining

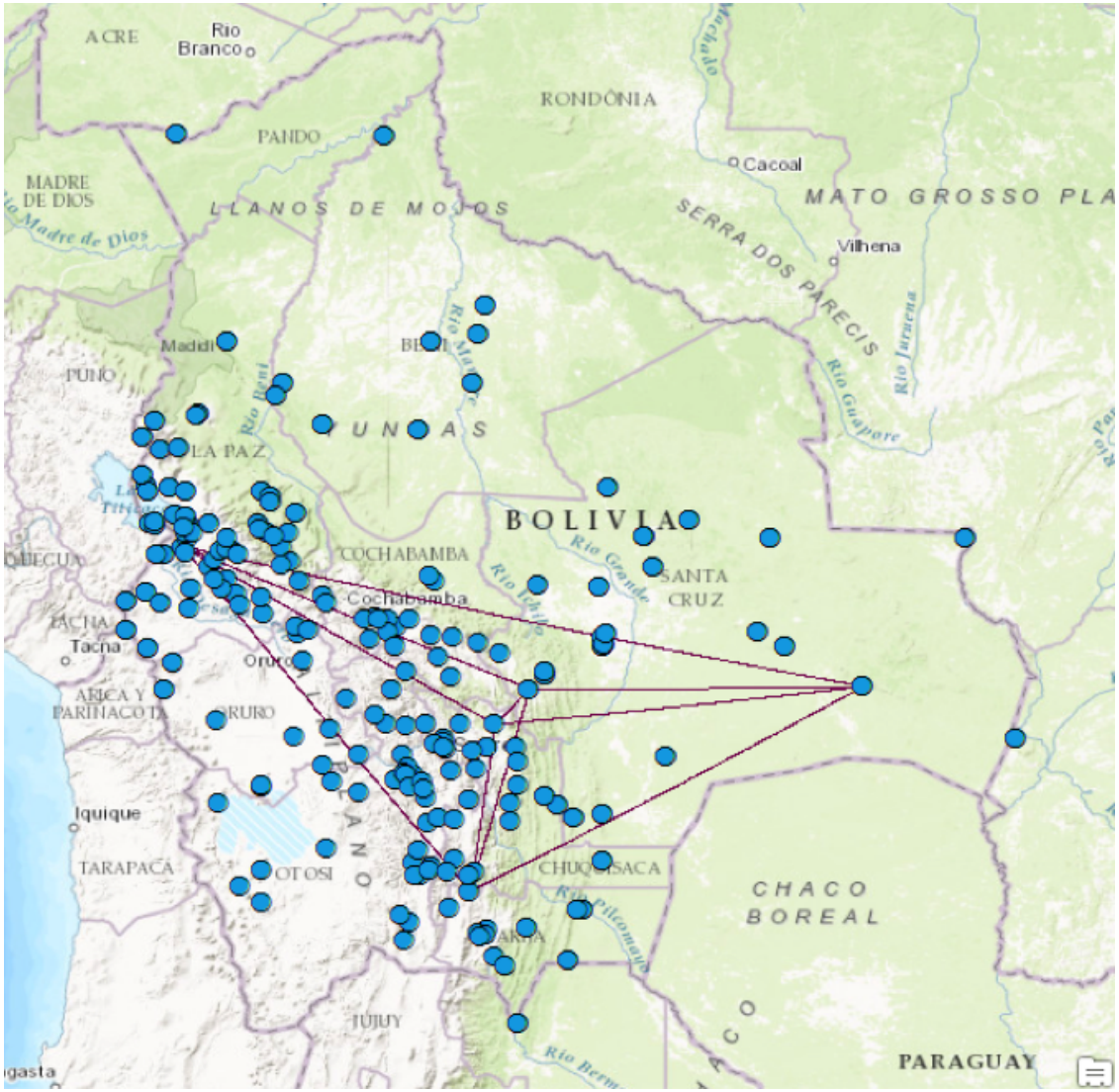


An open pit mine is subdivided into blocks. A network $G = (N, A)$:

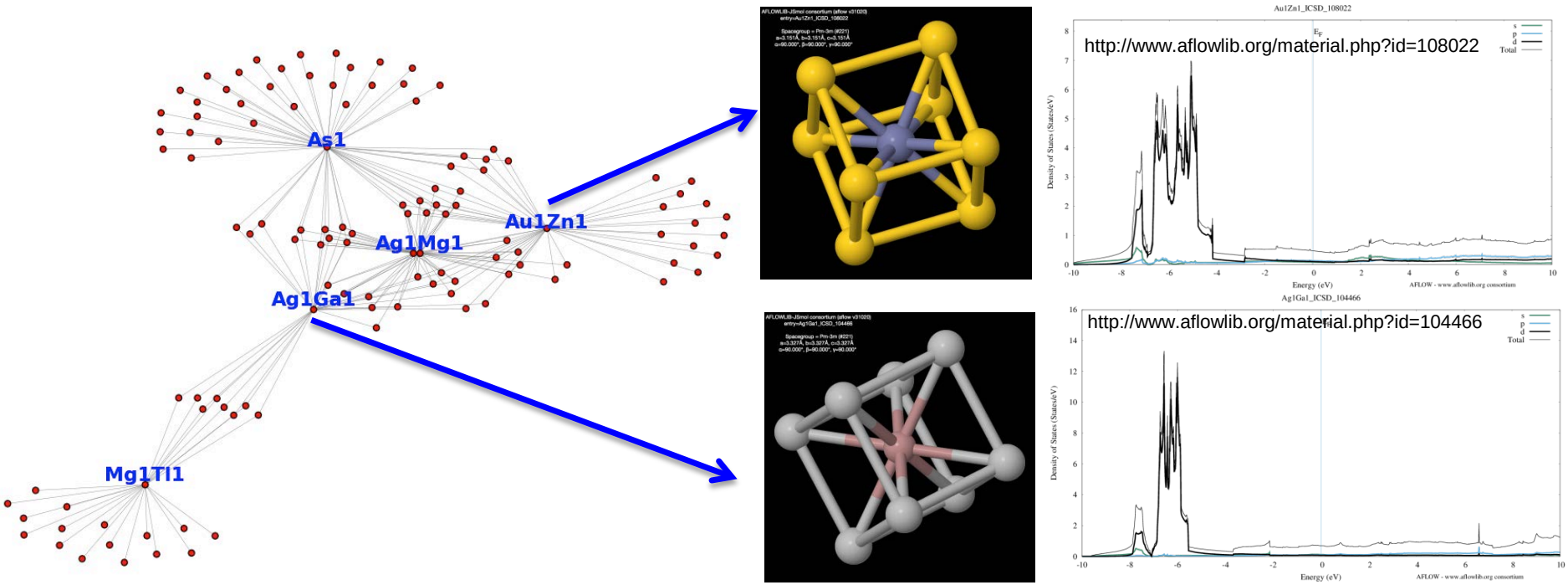
- Each block is described by a node $i \in N$
  - w_i =net revenue from block i (ore value $-$ processing cost).
- If block j must be removed before block i then $(i, j) \in A$

To maximize the profit we need to find a maximum weight closure in G (a subset of nodes with no leaving arcs).

Wind farm location

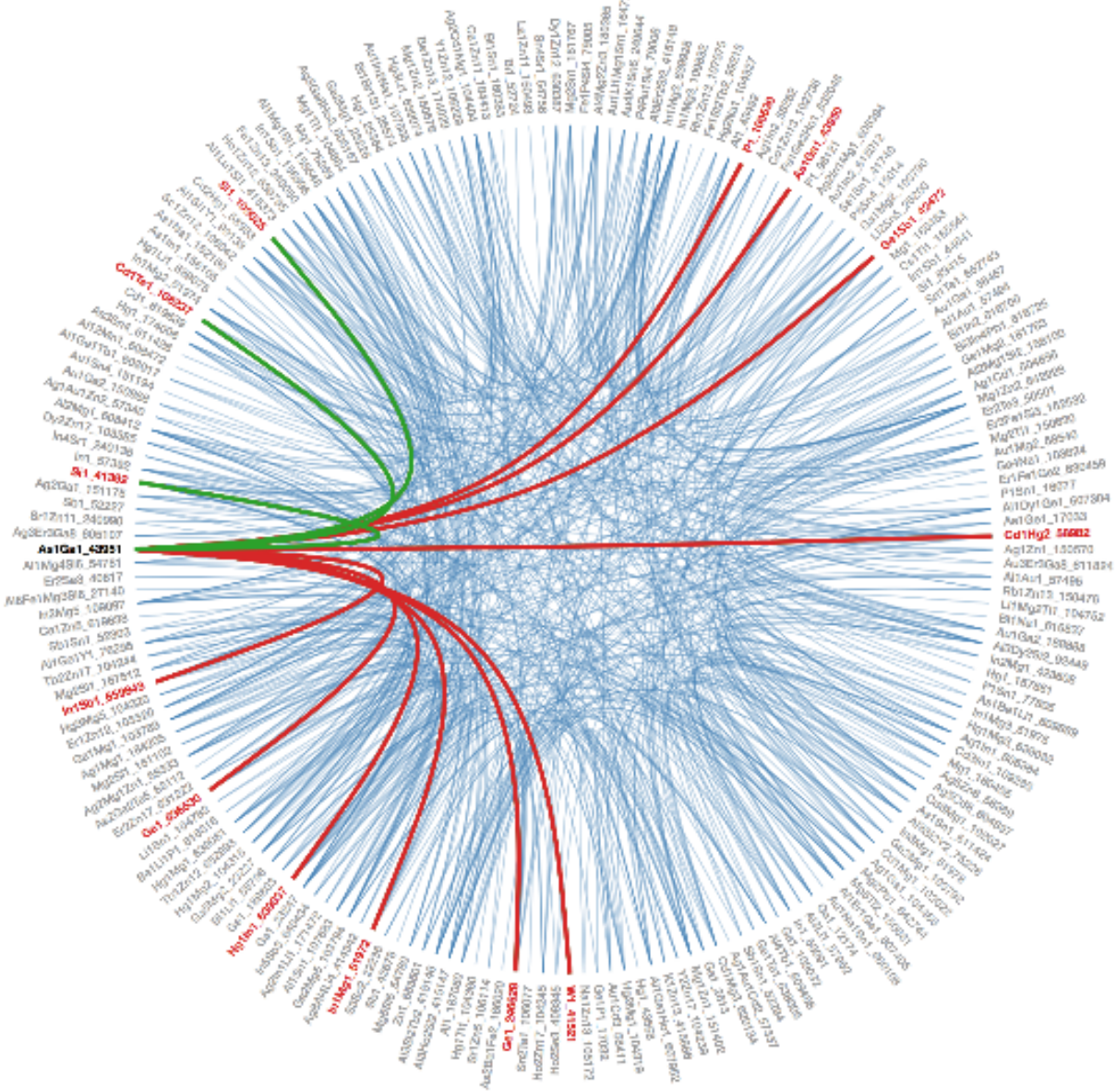


Networks of materials

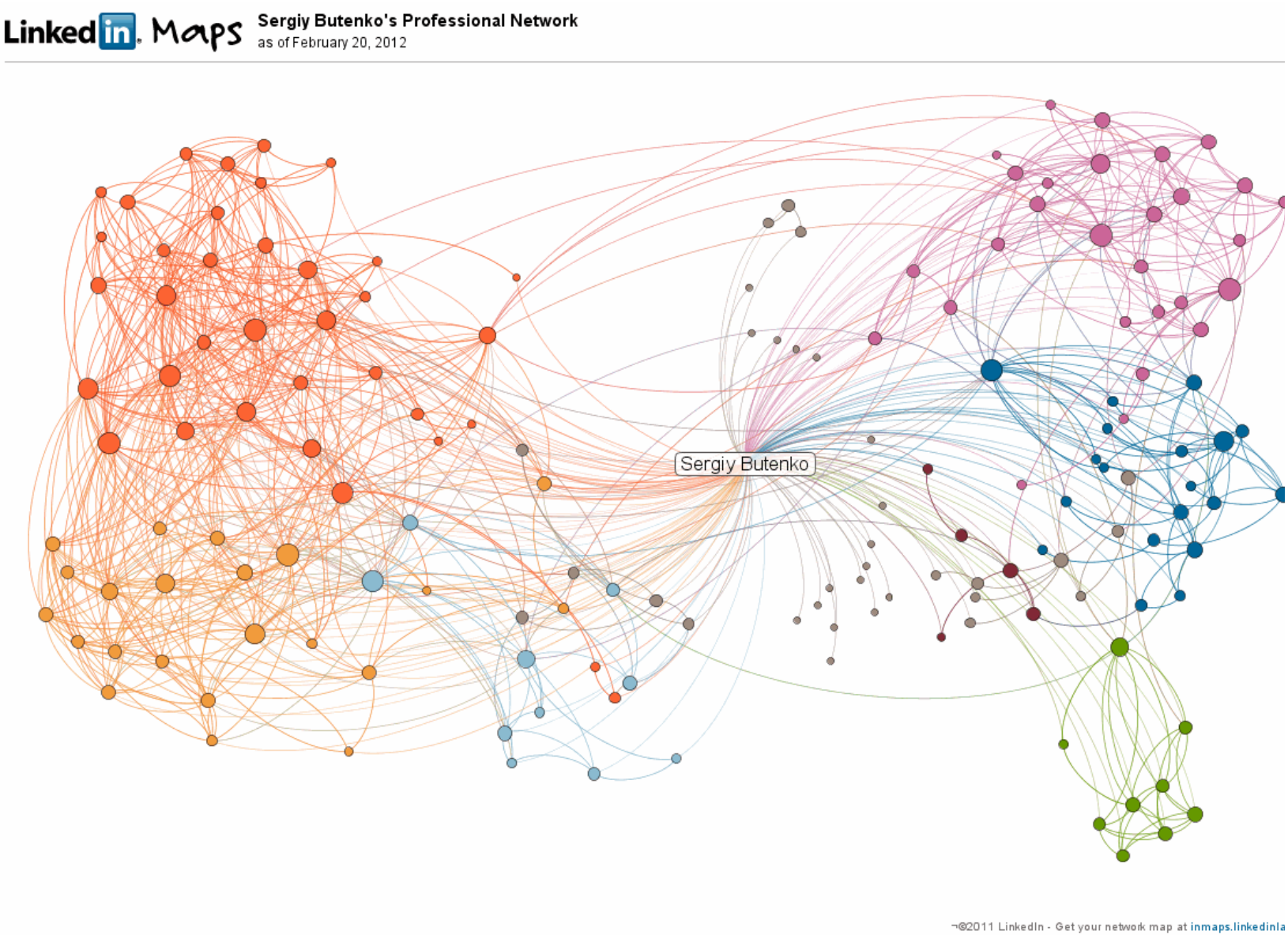


A. Veremyev, L. Liyanage, V. Boginski, M. Fornari, M. Buongiorno Nardelli, S. Curtarolo, and S. Butenko. Networks of materials. Working paper.

Networks of materials: D3.js interactive web visualization tool



Social networks



Outline

Social networks

A social network is described by $G = (V, E)$ where V is the set of "actors" and E is the set of "ties".

- actors are people and a tie exists if two people know each other.
- actors are wire transfer database records and a tie exists if two records have the same *matching field*.
- *Cohesive subgroups* are "closely knit groups" in a social network.
- *Social cohesion* is often used to explain and develop sociological theories.

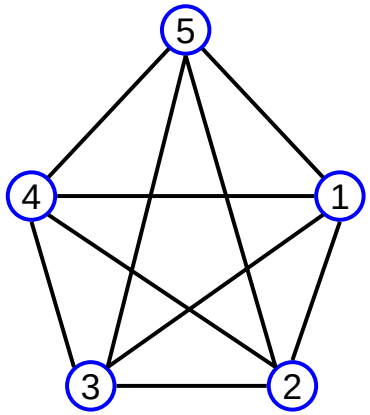
Properties of cohesive subgroups

Some desirable properties of a cohesive subgroup are:

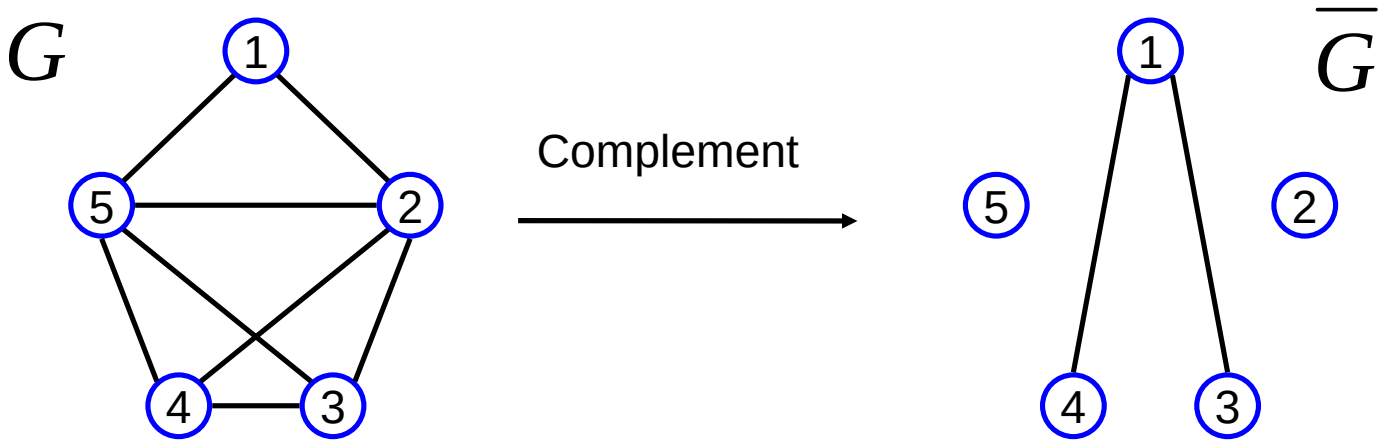
- Familiarity (degree);
- Reachability (distance, diameter);
- Robustness (connectivity);
- Density (edge density).

Cliques

- *Etymology:* The term *clique* originates from Old French *cliquer* meaning *make a noise*
- *WordNet dictionary definition:* an exclusive circle of people with a common purpose
- Luce and Perry (1949): social clique – a group of people that know (are friends of) all other people in the group



Cliques and independent sets



{1,2,5} : maximal clique

{1,4} : maximal independent set

{2,3,4,5} : maximum clique

{1,2,5} : maximal independent set

{1,4} : maximal clique

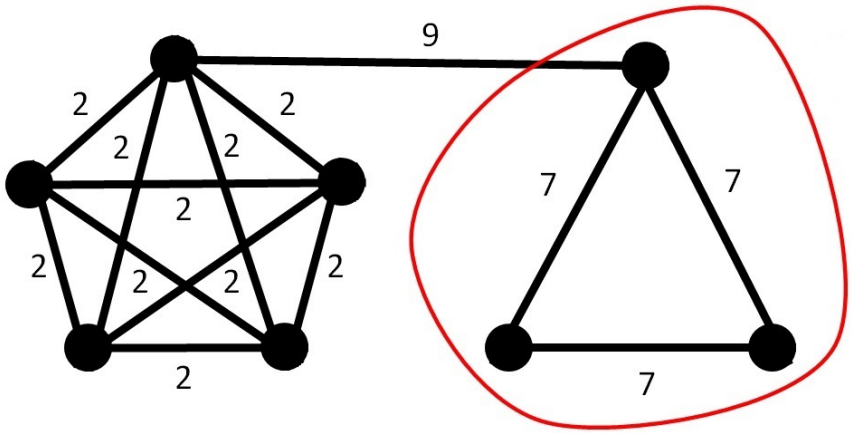
{2,3,4,5} : maximum independent set

Maximum edge weight clique problem

Given an undirected and edge-weighted graph $G = (V, E)$, the MEWC problem seeks a clique with the maximum total weight.

$$C^* = \underset{C}{argmax} \sum_{(i,j) \in E(C)} w_{ij}$$

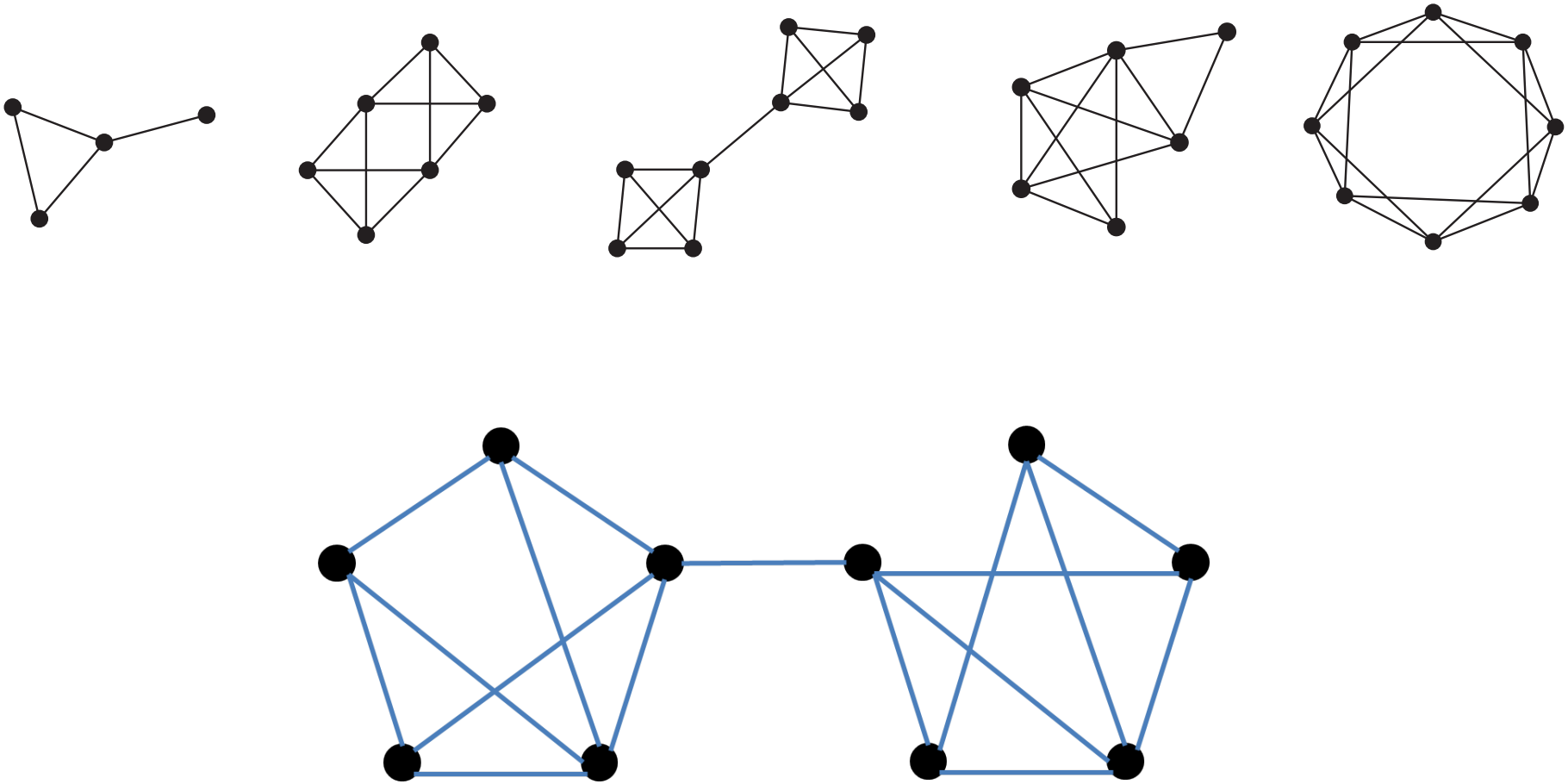
where, $w_{ij} > 0$ is the weight associated with edge (i, j).



Clusters in real-life networks

Cliques may be overly restrictive for practical purposes

1xg0 (immune sys.) 1p9m (signaling) 1dxr (photosynthesis) 1ruz (viral protein) 1kw6 (lyase)



Alternatives to clique

$G = (V, E)$. $S \subseteq V$ is

- s-**clique** if $d_G(v, v') \leq s$, for any $v, v' \in S$ (Luce 1950)
- s-**club** if $diam(G[S]) \leq s$ (Alba 1973, Mokken 1979)
- s-**plex** if $\delta(G[S]) \geq |S| - s$ (Seidman & Foster 1978)
- s-**defective clique** if $G[S]$ has at least $\binom{|S|}{2} - s$ edges (Yu et al. 2006)
- k-**core** if $\delta(G[S]) \geq k$ (Seidman 1983)
- k-**block** if $\kappa(G[S]) \geq k$ (Moody & White 2003)
- γ-**quasi-clique** if $\rho(G[S]) \geq \gamma$ (Abello et al. 2002)
- (λ, γ)-**quasi-clique** if $\delta(G[S]) \geq \lambda(|S| - 1)$ and $\rho(G[S]) \geq \gamma$ (Brunato et al. 2008)
- ...

Outline

"The whole is more than the sum of its parts."
–Aristotle (384-322 BC)

Alternative clique definitions

(a) Vertices are **distance one** away from each other

(b) Vertices induce a subgraph of **diameter one**

(c) Every **one** vertex forms a **dominating set**

Alternative clique definitions

(d) **Degree:** Each vertex neighbors **all** vertices

(e) **Density:** Vertices induce a subgraph that has **all** possible edges

(f) **Connectivity:** need to be remove **all** vertices to obtain a disconnected induced subgraph

Elementary clique-defining properties

Proposition

A subset of vertices C is a clique in G if and only if one of the following conditions hold:

a) *Pairwise distances:* $d_G(v, v') = 1$, *for any* $v, v' \in C$;

b) *Diameter:* $diam(G[C]) = 1$;

c) *Domination:* $D = \{v\}$ *is a dominating set in* $G[C]$, *for any* $v \in C$;

d) *Minimum degree:* $\delta(G[C]) = |C| - 1$;

e) *Edge density:* $\rho(G[C]) = 1$;

f) *Vertex connectivity:* $\kappa(G[C]) = |C| - 1$.

Defining clique relaxations

We can define clique relaxations by

(i) **restricting** a **violation** of an elementary clique-defining property or by

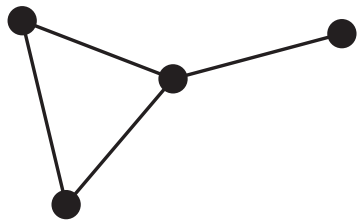
(ii) **ensuring** the **presence** of an *elementary clique-defining property*

(i) Restricting a violation

Pairs of vertices are **distance at most** s away from each other – s-**clique**

Induced subgraph is of **diameter at most** s – s-**club**

1xg0 (immune sys.)

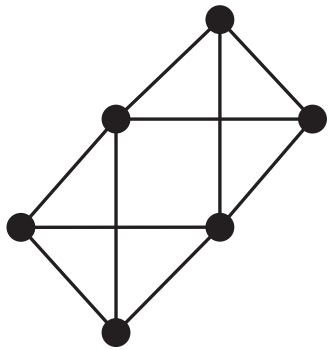


2-club

(i) Restricting a violation

Any set of size s ensures **domination – s-plex**

1p9m (signaling)



3-plex

(i) Restricting a violation

We replaced **one** with **at most** s in the alternative clique definitions

We can also replace **all** with **all but** s

(i) Restricting a violation

Degree: Each vertex neighbors **all but** s vertices – s-**plex** again

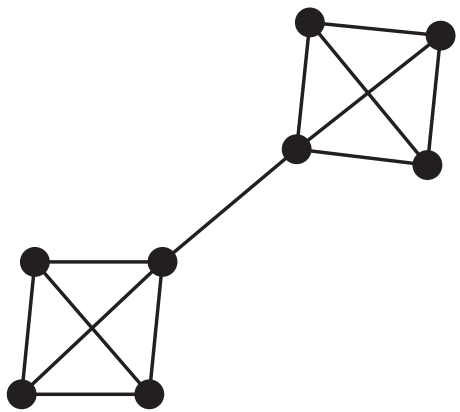
Density: Vertices induce a subgraph that has **all but** s possible edges – s-**defective clique**

Connectivity: need to be remove **all but** s vertices to obtain a disconnected induced subgraph – s-**bundle**

(ii) Ensuring a property

Each vertex has **degree at least k – k-core**

1dxr (photosynthesis)



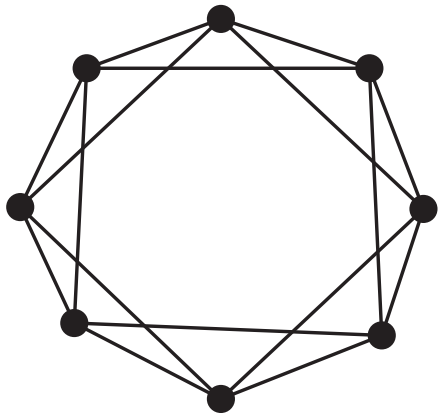
3-core

(ii) Ensuring a property

At least k vertices need to be removed

to disconnect the induced subgraph – k-**block**

1kw6 (lyase)

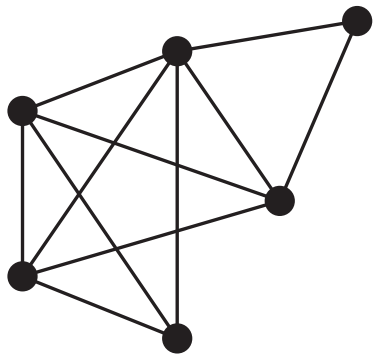


4-connected

Relative relaxations

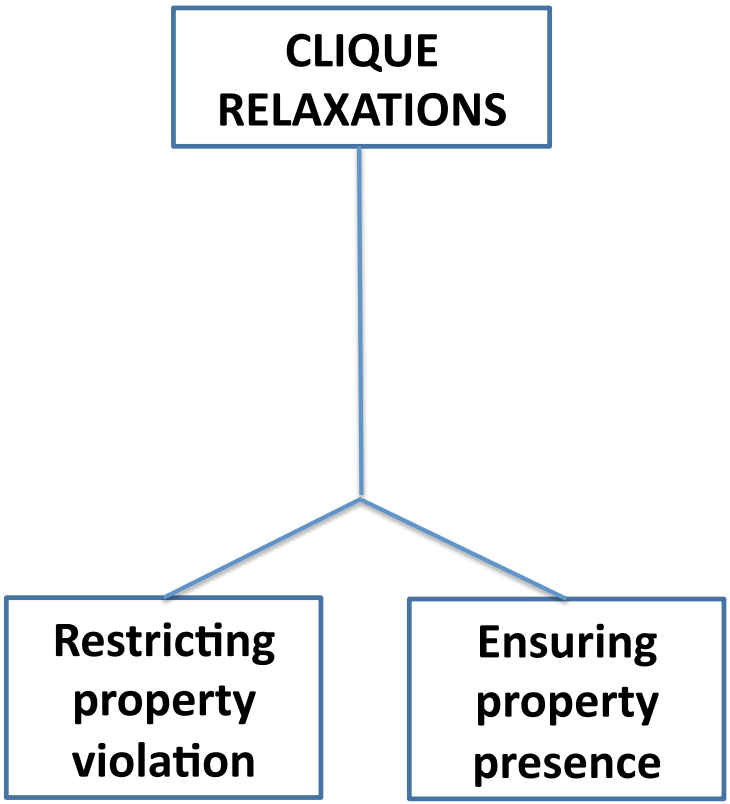
Vertices induce a subgraph that has **the fraction** γ of all possible edges – **γ-quasi-clique**

1ruz (viral protein)

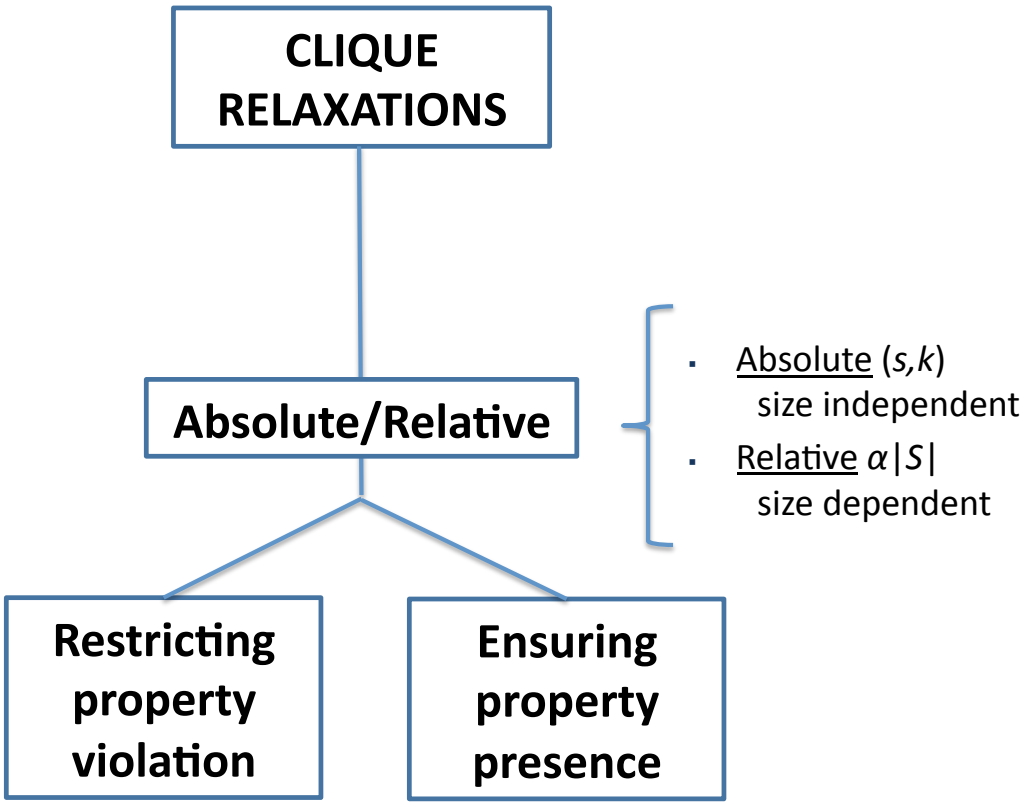


.7-quasiclique

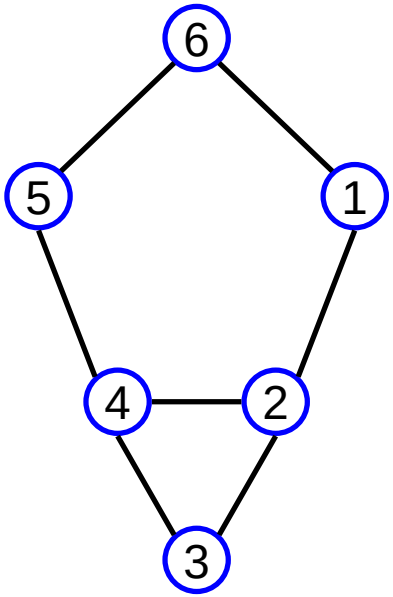
Nature of a clique relaxation



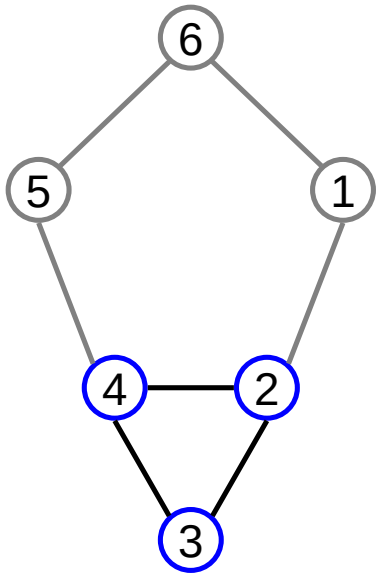
Nature of a clique relaxation



s-clique vs s-club

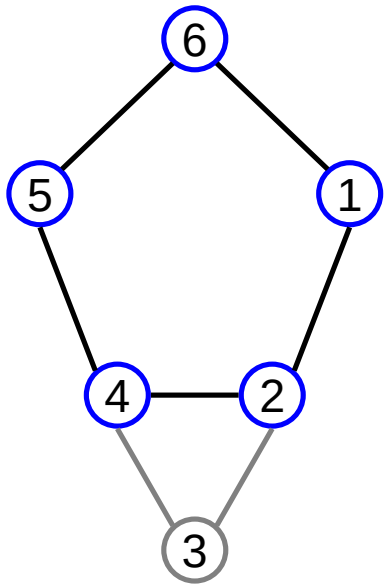


s-clique vs s-club



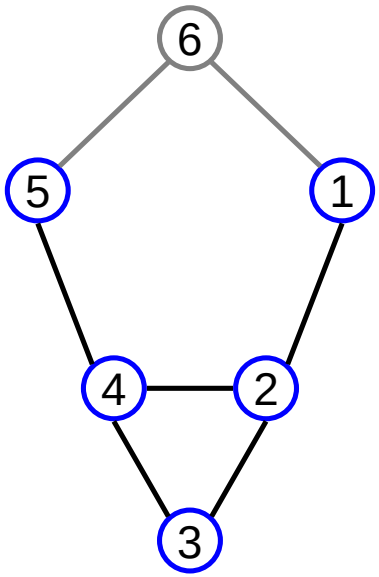
- $\{$**2,3,4**$\}$ is a 1-club ... the "regular" clique

s-clique vs s-club



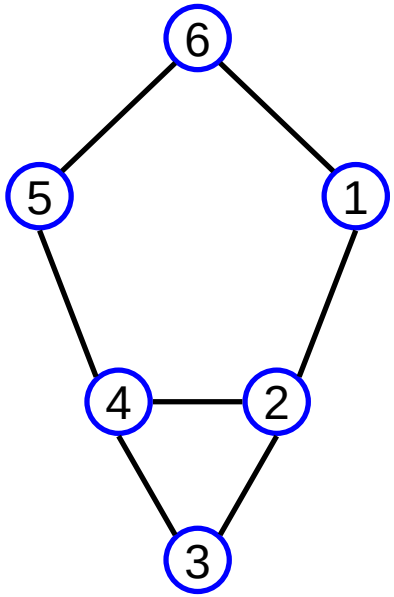
- $\{2,3,4\}$ is a 1-club ... the "regular" clique
- $\{\mathbf{1,2,4,5,6}\}$ is a 2-club

s-clique vs s-club



- $\{2,3,4\}$ is a 1-club ... the "regular" clique
- $\{1,2,4,5,6\}$ is a 2-club
- $\{\mathbf{1,2,3,4,5}\}$ is a 2-clique but NOT a 2-club

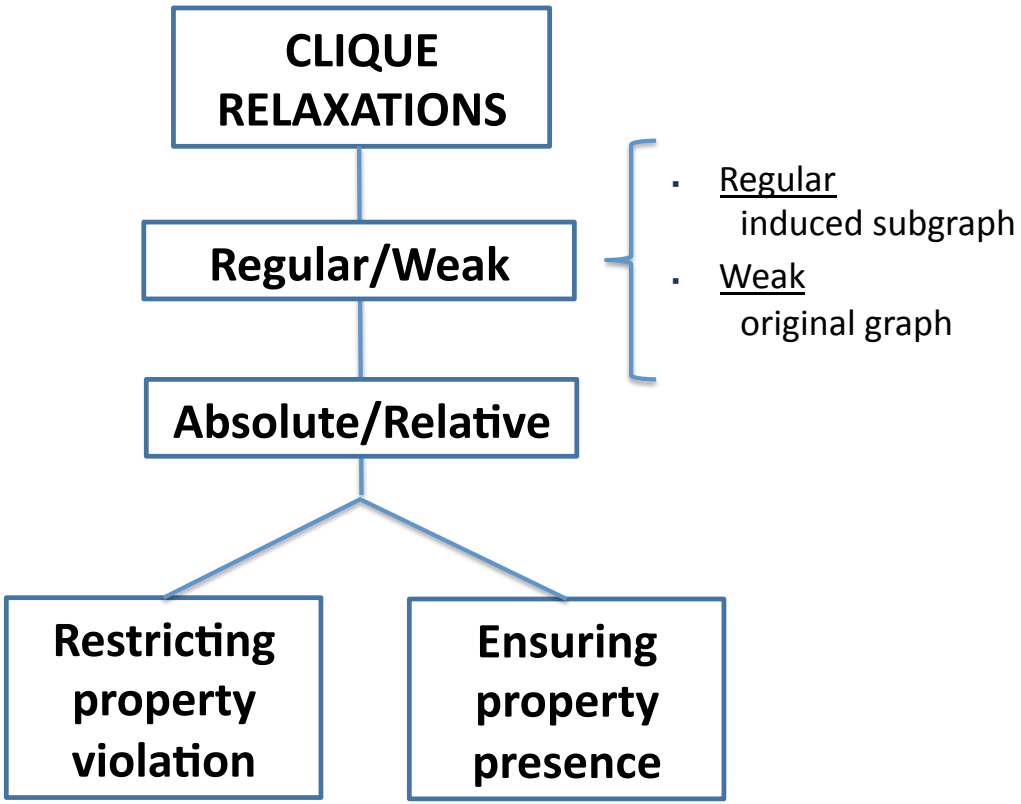
s-clique vs s-club



- $\{2,3,4\}$ is a 1-club ... the "regular" clique
- $\{1,2,4,5,6\}$ is a 2-club
- $\{1,2,3,4,5\}$ is a 2-clique but NOT a 2-club
- **maximality** of a 2-club is harder to test

s-clique appears to be a **weaker** cluster than s-club

Nature of a clique relaxation



Weak clique relaxations

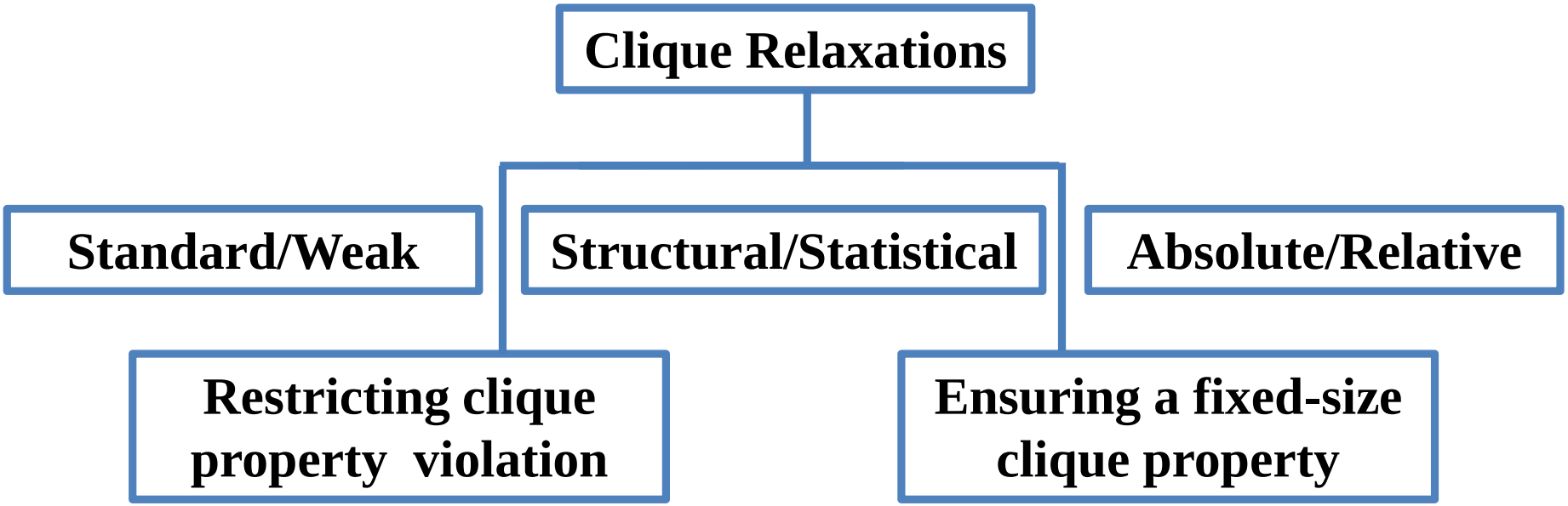
Distance-based: s-**clique (weak** s-**club)**

Vertices in S are **distance at most** s away from each other **in** G.

Connectivity-based: **weak** k-**block**

Any two vertices in S have **at least** k vertex-independent **paths** between them **in** G

Clique relaxations taxonomy



J. Pattillo, N. Youssef, and S. Butenko. On clique relaxation models in network analysis. *European Jour. of Oper. Res.*, 226: 9–18, 2013.

Order of a clique relaxation

It may be useful to relax **more than one** elementary clique-defining property

- **Clique** is the only clique relaxation of order 0
- Clique relaxations of **first order** relax **one** of the elementary properties (distance, diameter, ...) used to define clique
- Clique relaxations of **second order** relax **two** of the elementary clique-defining properties
- ...

Higher order clique relaxations

Simple Higher Order Relaxations: relaxing multiple elementary clique-defining properties simultaneously

(λ, γ)-*quasiclique*: Each vertex is connected to *at least* $\lambda(|S| - 1)$ vertices, and the induced subgraph has at least *the fraction* γ of all possible edges.

Robust Higher Order Relaxations: connectivity *embedded* into the definition (k-robustness/k-heredity)

k-*robust* s-*club*: The induced subgraph is not only an s-club, but also the removal of up to k vertices still preserves the s-club property.

Additional elementary clique-defining properties

A subset of vertices C is a clique in G if and only if one of the following conditions hold:

g) Independence number $\alpha(G[C]) = 1$;

h) Vertex cover number $\tau(G[C]) = |C| - 1$;

i) Chromatic number $\chi(G[C]) = |C|$;

j) Clique cover number $\bar{\chi}(G[C]) = 1$;

k) Edge connectivity number $\lambda(G[C]) = |C| - 1$.

Canonical clique relaxations

	Reachability	Familiarity		Composition	Robustness
	Diameter	Domination[1]	Density[2]	Degree[3]	Connectivity[4]
Clique	"one"	"one"	"all"	"all"	"all"
s-club	"at most s"				
s-plex		"s "			
γ-quasiclique			"at least γ"		
k-core				"at least k"	
k-connected					"at least k"

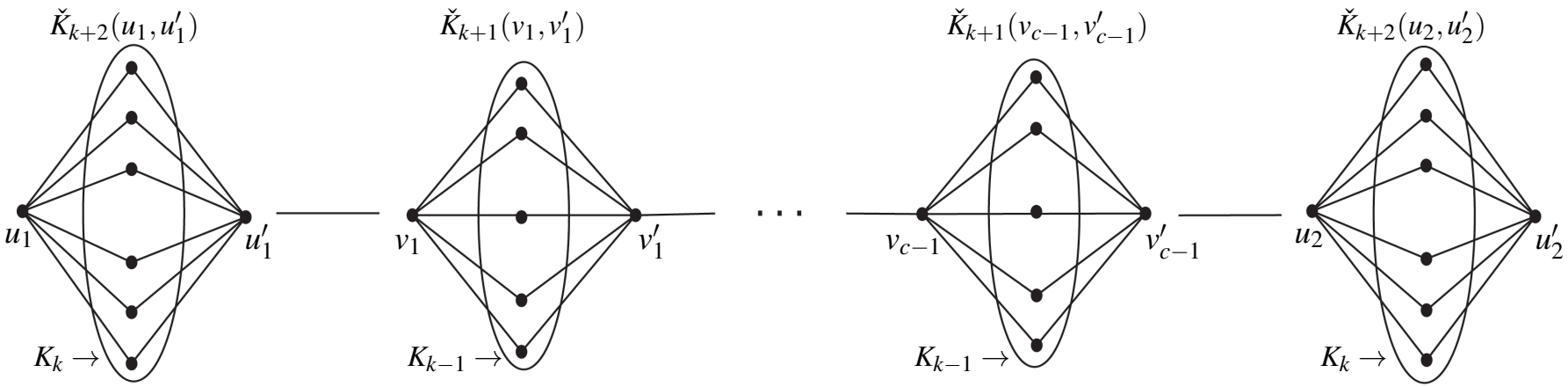
[1] *All vertices are dominated by*, [2] *Edges included*, [3] *Every vertex is connected to*, [4] *To disconnect, remove*

Diameter of k-cores

Let S be a k-core in G. If $G[S]$ is connected then $diam(G[S]) \leq d'_k$, where

$$d'_k = \max\left\{\left\lceil \frac{|S|}{k+1} \right\rceil, 3\left(\left\lfloor \frac{|S|-z}{k+1} \right\rfloor - 1\right) + z, z \in \{0,1,2\}\right\}$$

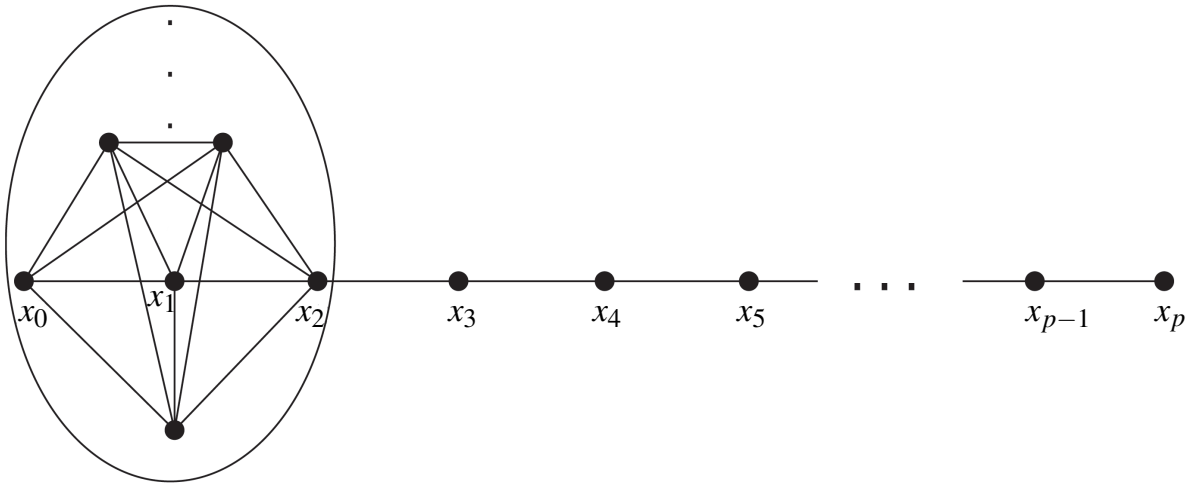
This bound is sharp



Diameter of γ-quasi-cliques

Let S be a γ-quasi-clique in G. If $G[S]$ is connected, then $diam(G[S]) \leq d_\gamma$, where

$$d_\gamma = \left\lfloor |S| + \frac{1}{2} - \sqrt{\gamma|S|^2 - (2+\gamma)|S| + \frac{17}{4}} \right\rfloor .$$



K_q without edge (x_0, x_2)

Cohesiveness properties

$S \subseteq V$	Diameter	Dominating Set	Minimum Degree	Edge Density	Connectivity
Clique	"one"	"one"	"all"	"one"	"all"
s-club	s	$\lvert S\rvert - 1$	1	$\frac{2}{\lvert S\rvert}$	1
s-plex	s	s	$\lvert S\rvert - s$	$1 - \frac{s-1}{\lvert S\rvert - 1}$	$\lvert S\rvert - 2s + 2$
k-core	d'_k	$\lvert S\rvert - k$	k	$\frac{k}{\lvert S\rvert - 1}$	$2k + 2 - \lvert S\rvert$
γ-quasi-clique	d_γ	$\lvert S\rvert$	$\left\lceil \gamma\binom{\lvert S\rvert}{2} - \binom{\lvert S\rvert - 1}{2} \right\rceil$	γ	$\left\lceil \gamma\binom{\lvert S\rvert}{2} - \binom{\lvert S\rvert - 1}{2} \right\rceil$
k-block	$\left\lfloor \frac{\lvert S\rvert - 2}{k} + 1 \right\rfloor$	$\lvert S\rvert - k$	k	$\frac{k}{\lvert S\rvert - 1}$	k

Outline

Optimization problems

Let RELAXED CLIQUE (RC)refer to a subset of vertices that satisfies the definition of an arbitrary clique relaxation concept.

Definition

- A subset of vertices S is called a maximal RC if it is a RC and is not a proper subset of a larger RC.
- A subset of vertices S is called a maximum RC if there is no larger RC in the graph.
- The maximum RC problem asks to compute a maximum RC in the graph, and the size of a maximum RC is called the RC number.

Structiural properties

Definition (Heredity)

A graph property Π is said to be *hereditary on induced subgraphs*, if for any graph G with property Π the deletion of any subset of vertices does not produce a graph violating Π.

Definition (Weak heredity)

A graph property Π is said to be *weakly hereditary*, if for any graph $G = (V, E)$ with property Π all subsets of V demonstrate the property Π in G.

Structiural properties

Definition (Quasi-heredity)

A graph property Π is said to be *quasi-hereditary*, if for any graph $G = (V, E)$ with property Π and for any size $0 \leq r < |V|$, there exists some subset $R \subset S$ with $|R| = r$, such that $G[S \setminus R]$ demonstrates property Π.

Definition (k-Heredity)

A graph property Π is said to be *k-hereditary on induced subgraphs*, if for any graph G with property Π the deletion of any subset of vertices with up to k vertices does not produce a graph violating Π.

Hereditary clique relaxations

Models restricting the violation of an elementary clique-defining property ("all but s" type):

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

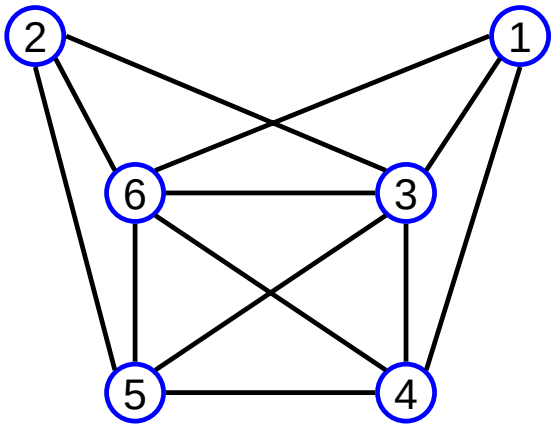
Combinatorial branch-and-bound algorithms available for the maximum clique problem can be easily adopted for these models.

Hereditary clique relaxations: s-plex

Definition

A subset of vertices S is an s-plex if the minimum degree in the induced subgraph $\delta(G[S]) \geq |S| - s$

i.e., every vertex in $G[\mathbf{S}]$ has degree at least $|\mathbf{S}| - s$.

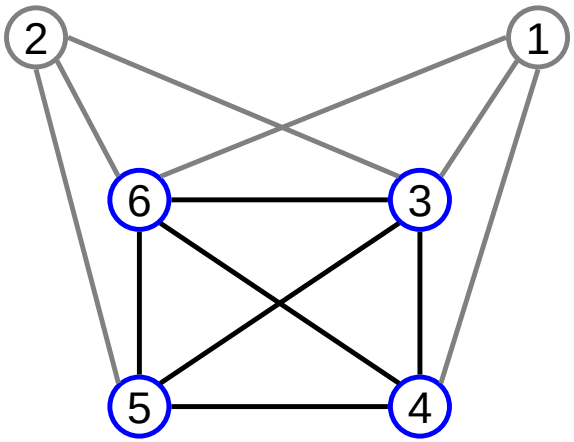


Hereditary clique relaxations: s-plex

Definition

A subset of vertices S is an s-plex if the minimum degree in the induced subgraph $\delta(G[S]) \geq |S| - s$

i.e., every vertex in $G[\mathbf{S}]$ has degree at least $|\mathbf{S}| - s$.



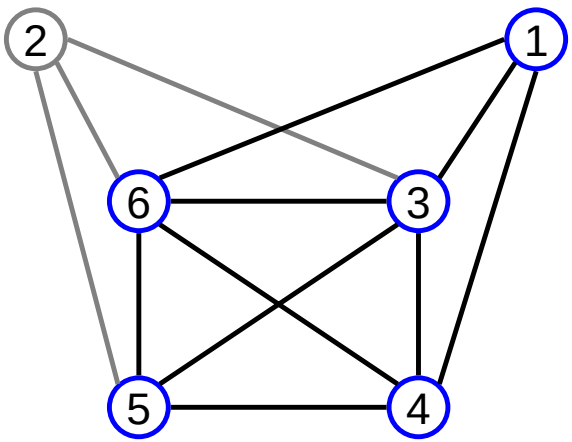
- $\{\mathbf{3,4,5,6}\}$ is a 1-plex ... the "regular" clique

Hereditary clique relaxations: s-plex

Definition

A subset of vertices S is an s-plex if the minimum degree in the induced subgraph $\delta(G[S]) \geq |S| - s$

i.e., every vertex in $G[\mathbf{S}]$ has degree at least $|\mathbf{S}| - s$.



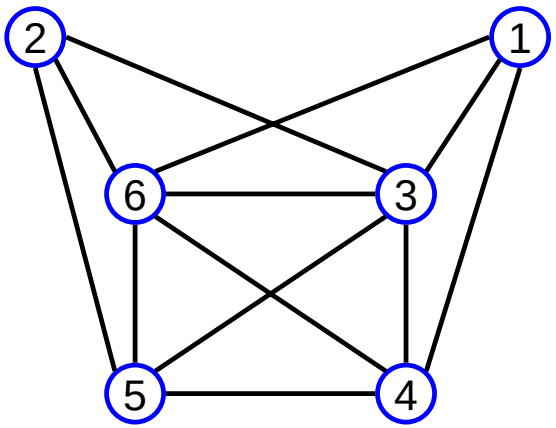
- $\{3,4,5,6\}$ is a 1-plex ... the "regular" clique
- $\{\mathbf{1,3,4,5,6}\}$ is a 2-plex (and NOT a 1-plex)

Hereditary clique relaxations: s-plex

Definition

A subset of vertices S is an s-plex if the minimum degree in the induced subgraph $\delta(G[S]) \geq |S| - s$

i.e., every vertex in $G[\mathbf{S}]$ has degree at least $|\mathbf{S}| - s$.



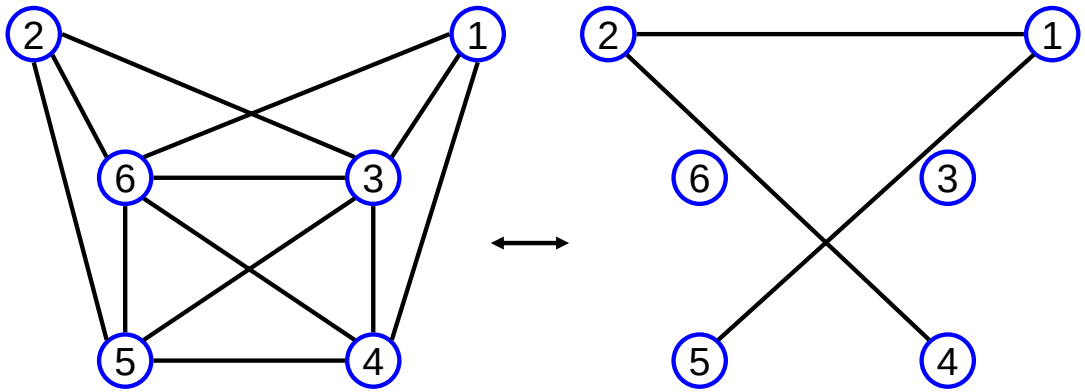
- $\{3,4,5,6\}$ is a 1-plex ... the "regular" clique
- $\{1,3,4,5,6\}$ is a 2-plex (and NOT a 1-plex)
- $\{\mathbf{1,2,3,4,5,6}\}$ is a 3-plex (and NOT a 2-plex)

Complementary structure: co-s-plex

Definition

A subset of vertices S is a co-s-plex if the maximum degree in the induced subgraph $\Delta(G[S]) \leq s - 1$.

i.e., degree of every vertex in $G[S]$ is at most $s - 1$.



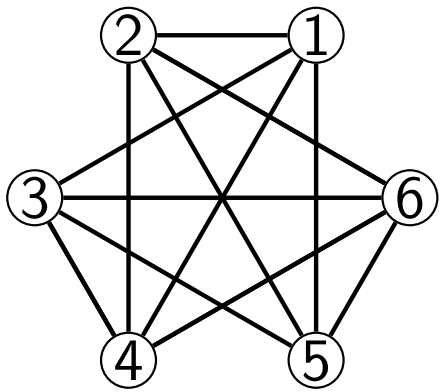
3-plex Co-3-plex

Hereditary clique relaxations: s-defective clique

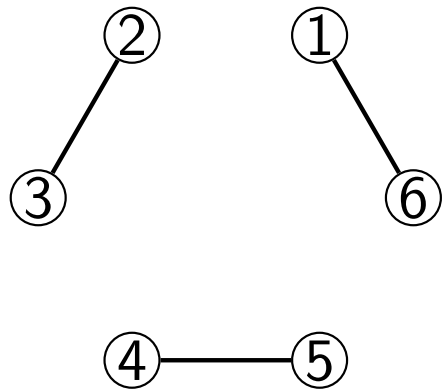
Definition

A subset of vertices S is an s-defective clique if the number of edges in the induced subgraph is at least $\binom{|S|}{2} - s$.

i.e., at most s edges are missing.



3-defective clique



co-3-defective clique

Structural properties of an s-plex

If G is an s-plex then

1. Every subgraph of G is a s-plex;
2. If $s < \frac{n+2}{2}$ then $diam(G) \leq 2$;
3. $\kappa(G) \geq n - 2s + 2$.
4. Any s vertices in G form a *dominating set* in G.

- s-plexes for "small" s values, guarantee reachability and connectivity while relaxing familiarity.

Hereditary clique relaxations

- The *maximum* Π *problem* is to find the largest order induced subgraph that does not violate property Π
- Π is said to be *nontrivial* if it is true for a single vertex graph and is not satisfied by every graph
- Π is said to be *interesting* if there are arbitrarily large graphs satisfying Π

Theorem (Yannakakis, 1978)

The maximum Π *problem for nontrivial, interesting graph properties that are hereditary on induced subgraphs is NP-hard.*

Weakly hereditary clique relaxations

Weak clique relaxations:

- s-clique
- weak k-block

Optimization problems for these models can be reduced to maximum clique in corresponding auxiliary graphs.

Quasi-hereditary clique relaxations

- Quasi-clique

The maximum quasi-clique problem is extremely challenging to solve exactly.
Quasi-heredity suggests that greedy randomized heuristics should be effective in practice.

Exact algorithms

- Integer programming
  - B. Balasundaram, S. Butenko, and I. Hicks. Clique relaxations in social network analysis: the maximum k-plex problem. *Operations Research*, 2011.
  - A. Veremyev, O. A. Prokopyev, S. Butenko, and E. L. Pasiliao. Exact MIP-based approaches for finding maximum quasi-cliques and dense subgraphs. *Computational Optimization and Applications*, 2017.
- Combinatorial branch and bound
  - S. Trukhanov, C. Balasubramaniam, B. Balasundaram, and S. Butenko. Algorithms for detecting optimal hereditary structures in graphs, with application to clique relaxations. *Computational Optimization and Applications*, 2013.

Scale-reduction techniques

Using scale reduction techniques based on clique relaxations in conjunction with Östergård's max clique algorithm the clique number was obtained on all graphs in the SNAP database and 10-th DIMACS Implementation Challenge (graphs with up to ≈18.5 million vertices)

A. Verma, A. Buchanan, and S. Butenko. Solving the Maximum Clique and Vertex Coloring Problems on Very Large Sparse Networks. *INFORMS Journal on Computing*, 27: 164–177, 2015.

INFORMS Connect President's Pick for May 2015.

A. Buchanan, J. L. Walteros, B., and P. M. Pardalos. Solving maximum clique in sparse graphs: an $O(nm + 2^{d/4})$ algorithm for d-degenerate graphs. *Optimization Letters*, 8: 1611–1617, 2014.

Computing tight upper bounds

- The maximum clique problem is hard to approximate.
- Many effective heuristics have been developed, often yielding optimal solutions.
- Tight upper bounds are essential for proving optimality.
- Convex relaxations that determine upper bounds on $\omega(G)$ have been proposed.
  - Lovász "sandwich theorem": $\omega(G) \leq \vartheta(\bar{G}) \leq \chi(G)$.
- Computing such bounds involves lifting to spaces of very high dimensions, limiting their applicability in practice.

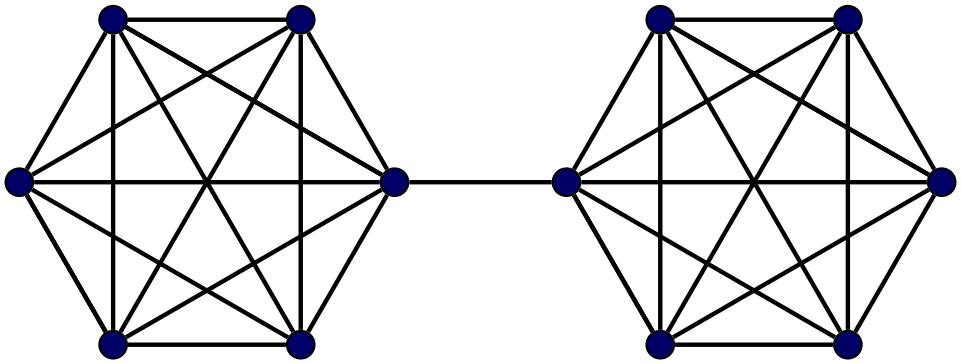
Computing tight upper bounds

- Consider a model Π defined by enforcing a property satisfied by a clique consisting of $k+1$ vertices.
- Solving the problem of minimizing the cardinality of a Π-set in G
  - yields a clique of cardinality $k+1$ whenever one exists, OR
  - Outputs an empty set or a set of size greater than $k+1$ if there is no clique of cardinality $k+1$ in the graph.
- To verify that $k = \omega(G)$ for a heuristic solution of cardinality k, we can consider the problem of minimizing the size of Π-set and
  - either detect infeasibility or
  - establish a nontrivial (that is, better than $k+1$) lower bound.

C. Balasubramaniam, B. Balasundaram, and S. Butenko. On upper bounds for the maximum clique problem. Working paper.

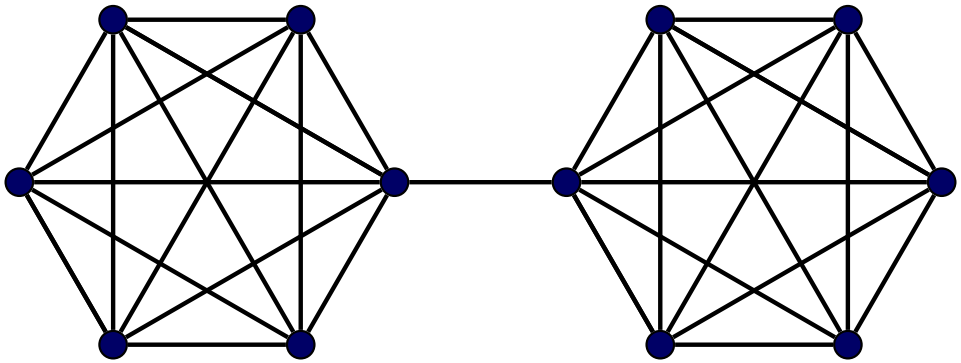
Unsupervised clustering

Partitioning vertices into ‘natural groups’ (clusters)



Unsupervised clustering

Partitioning vertices into 'natural groups' (clusters)



[HTML] **Community detection** in graphs

S Fortunato - Physics reports, 2010 - Elsevier

The modern science of networks has brought significant advances to our understanding of complex systems. One of the most relevant features of graphs representing real systems is community structure, or clustering, ie the organization of vertices in clusters, with many ...

☆ 99 Cited by 6655 Related articles All 47 versions Web of Science: 3233 »

Unsupervised clustering

Alternative approaches:

- Require each cluster to satisfy certain structural properties and minimize the number of clusters

  Minimum clique partitioning

- Optimize a certain quantitative measure of clustering quality

  Modularity: the fraction of edges that fall within clusters, minus the expected fraction of edges within clusters for a random graph with same degree distribution as the given network.

k-Community clustering

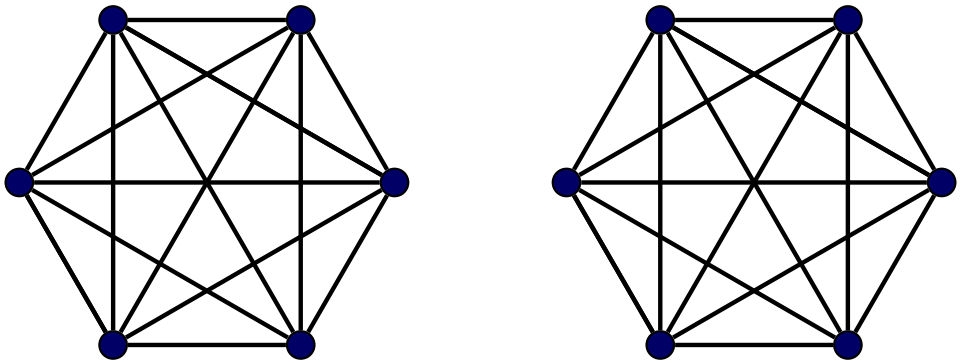
- Introduced a general purpose clustering algorithm based on clique relaxations.
- Do not aim to optimize any standard performance measure.
- Using k-community as a structure does well for a number of clustering quality measures.
- Enhancements to the basic algorithm can be designed according to requirements.

A. Verma and S. Butenko. Network clustering via clique relaxations: a community-based approach. In: *Graph Partitioning and Graph Clustering*. Ed. by D. A. Bader, H. Meyerhenke, P. Sanders, and D. Wagner. American Mathematical Society, 2013, pp.125–136.

Independent union of cliques

Definition (Independent union of cliques (IUC))

A subset of vertices C is called an independent union of cliques (IUC) if every connected component of $G[C]$ is a complete graph.



Independent union of cliques

Definition

An *open triangle* is a simple graph with three vertices and two edges (i.e., a 3-vertex path), and a *closed triangle* is a complete 3-vertex graph.

Proposition

$C \subseteq V$ is an IUC *if and only if no set of three vertices from C induces an open triangle in G.*

Clustering coefficients

- *Clustering coefficient* was introduced by Watts and Strogatz (1998) as a structural feature characterizing small-world networks
- "Two people are more likely to be friends if they have a friend in common"
- Noting that cohesive subgroups tend to have high clustering coefficients, we introduced a new clique relaxation, γ-*cluster*, based on the clustering coefficient

Z. Ertem, A. Veremyev, and S. Butenko. Detecting large cohesive subgroups with high clustering coefficients in social networks. *Social Networks* 46: 1–10, 2016.

Clustering coefficients

The **local clustering coefficient** C_i of node i of degree $d_G(i) \geq 2$ in G is given by

$$C_i = \frac{\sum_{j,k \in N_G(i), j<k} a_{jk}}{\binom{d_G(i)}{2}}.$$

The **global clustering coefficient** $\mathcal{C}$ of graph G that has at least one connected component with more than 2 vertices is given by

$$\mathcal{C} = \frac{\sum_{i \in V} \sum_{j,k \in N_G(i), j<k} a_{jk}}{\sum_{i \in V} \binom{d_G(i)}{2}}.$$

γ-Clusters

Given a graph $G = (V, E)$, a subset of vertices $C \subseteq V$ is called a

- **local γ-cluster** if $G[C]$ is connected and every node in C has the local clustering coefficient at least γ in $G[C]$;
- **global γ-cluster** if $G[C]$ is connected and $G[C]$ has the global clustering coefficient at least γ.

A local γ-cluster is also a global γ-cluster, whereas the converse does not hold in general

Defining clique using clustering coefficients

C is a clique if and only if it is a local/global γ-cluster with $\gamma = 1$.

Local γ-clustering algorithm

- Dropping the connectivity requirement from the definition of a γ-cluster may result in solutions corresponding to independent unions of γ-clusters
- These independent γ-clusters can be used as seeds for clustering methods
- For $\gamma = 1$ we obtain an independent union of cliques

Independent union of cliques

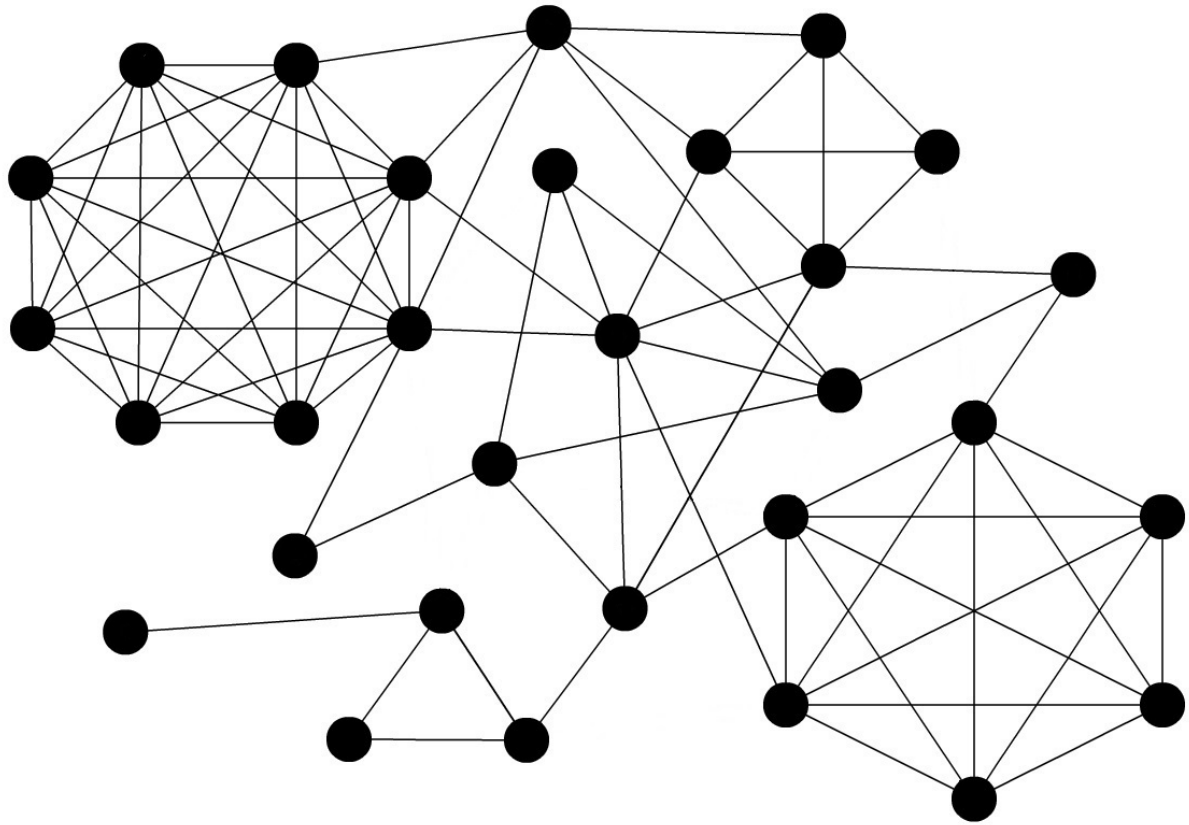
Definition

- A subset of vertices C is called an independent union of cliques (IUC) if every connected component of $G[C]$ is a complete graph.
- An IUC is maximal if it is not a subset of a larger IUC and maximum, if it there is no larger IUC in the graph.
- The maximum IUC problem is to find an IUC of maximum cardinality in G.
- This cardinality is denoted $\alpha^{\omega}(G)$ and called the IUC number of G.

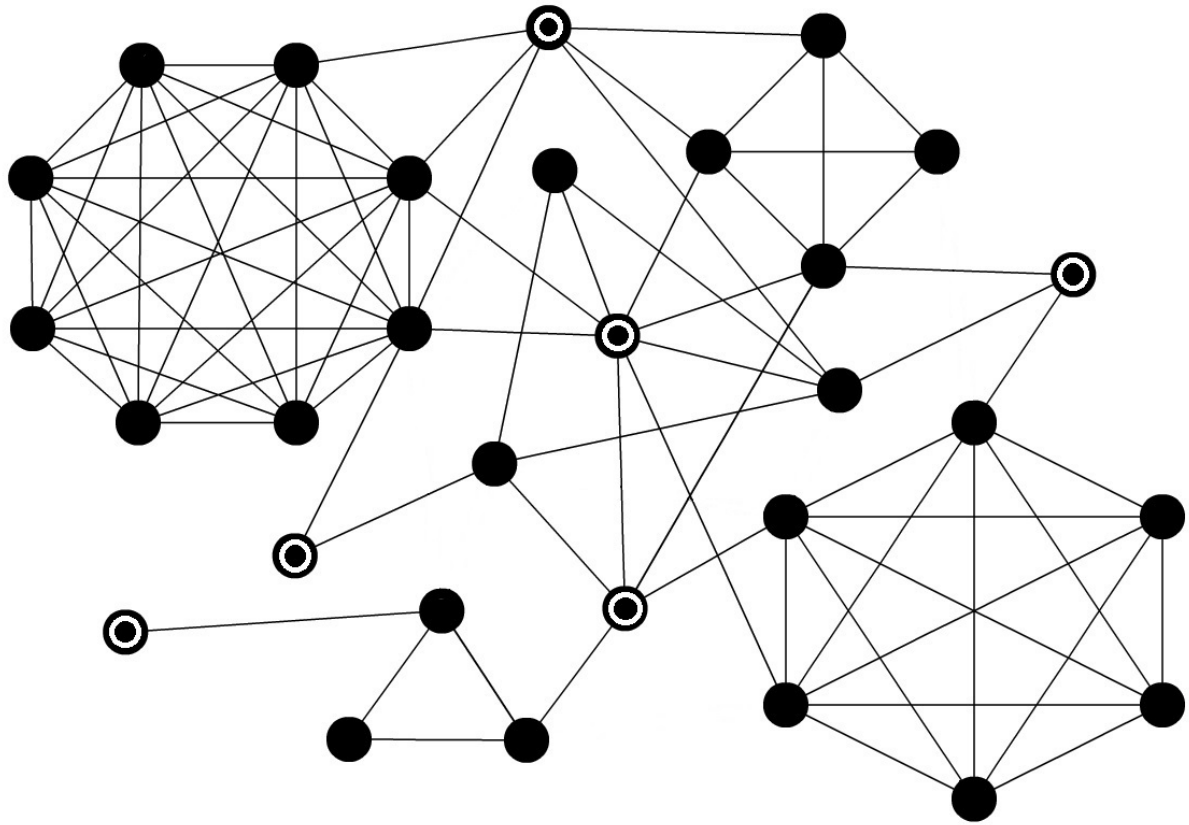
Independent union of cliques

- Maximum IUC is equivalent to the optimization version of *cluster vertex deletion problem* (Hüffner et al., 2010) (or the s-*plex cluster vertex deletion problem* with $s = 1$ (Bevern et al., 2012)
  - Find a minimum number of vertices that need to be removed from the graph so that the remaining vertices form an IUC.
  - D is an optimal solution of the cluster deletion problem if and only if $C = V \setminus D$ is a maximum IUC.

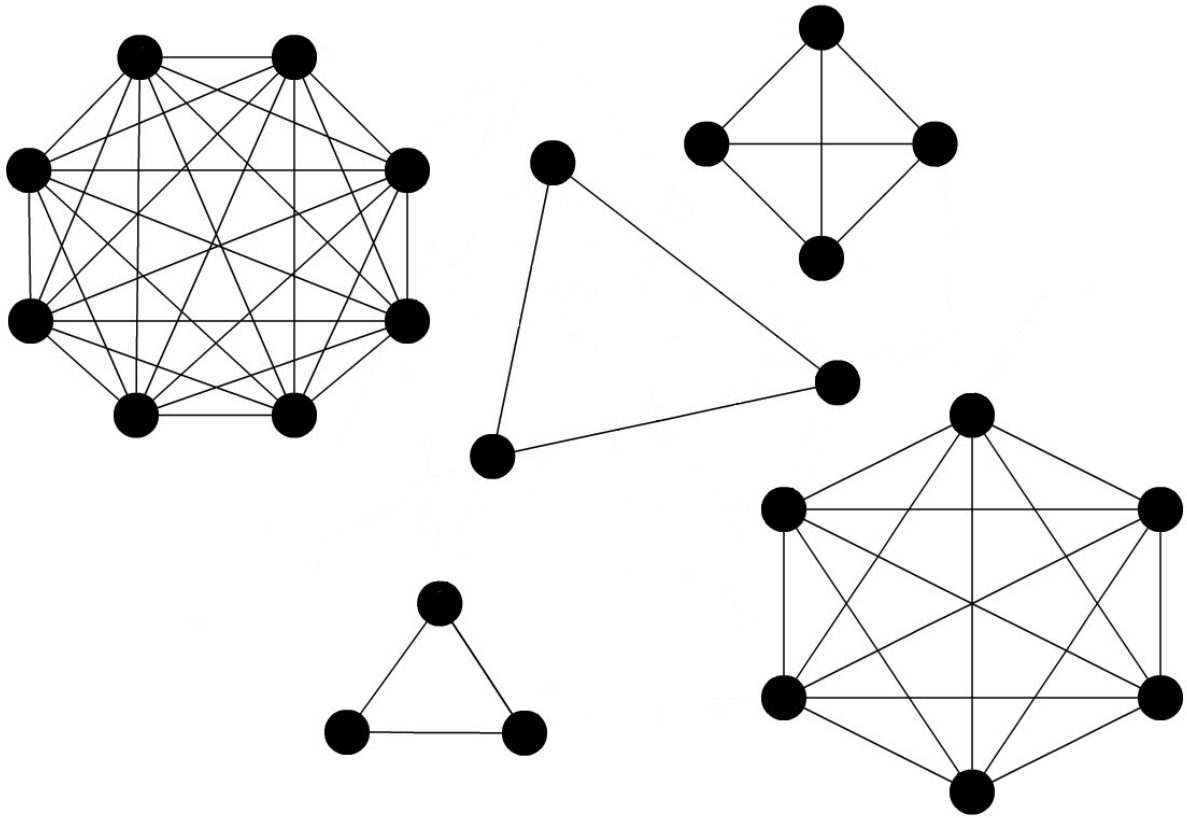
Independent union of cliques



Independent union of cliques



Independent union of cliques



cluster graph

Complementary structure: multi-partite clique

Definition

- A subset of vertices C is called a multi-partite clique (MPC) if it can be partitioned into $1 \leq k \leq |C|$ independent sets $C_1, \ldots, C_k$ such that any two vertices from different subsets are adjacent.
- The maximum MPC problem is to find an MPC of maximum cardinality, which is denoted by $\omega^{\alpha}(G)$ and is called the MPC number of G.

Fractional objective

Definition

Given a simple undirected graph $G = (V, E)$, where each vertex $i \in V$ is assigned two non-negative rational weights, a_i and b_i, the maximum ratio clique problem (MRCP) is to find a maximal clique C in G that maximizes the quantity $\frac{\sum_{i \in C} a_i}{\sum_{i \in C} b_i}$.

When $b_i = 1$ for all $i \in V$, we obtain a special case of MRCP, which we will refer to as the *maximum average weight clique problem* (MAWCP).


S. Sethuraman and S. Butenko. The maximum ratio clique problem. *Computational Management Science*, 12: 197–218, 2015.


O. Ursulenko, S. Butenko, and O. Prokopyev. A global optimization algorithm for solving the minimum multiple ratio spanning tree problem. *Journal of Global Optimization*, 56: 1029–1043, 2013.

"Best" approximation algorithms and heuristics

- For some problems there are hardness of approximation results stating that the problem is hard to approximate within a certain factor.
  - For example, the k-center problem is hard to approximate within a factor better than 2.
  - Then any polynomial-time algorithm approximating the k-center problem within the factor of 2 can be considered the "best" approximation algorithm for this problem.
- Maximum clique is hard to approximate within a factor $n^{1-\epsilon}$ for any positive ϵ.
- In this case, we call a heuristic "best" if it cannot be provably outperformed by any other polynomial-time algorithm (unless $P = NP$).

S. Kahruman-Anderoglu, A. Buchanan, S. Butenko, and O. Prokopyev. On provably best construction heuristics for hard combinatorial optimization problems. Networks 67: 238–245, 2016.

"Best" heuristics for k-club/clique

Theorem

Let positive integer constants k and l, $l < k$ be given. The problem of checking whether $\bar{\omega}_l(G) = \bar{\omega}_k(G)$ is NP-hard.

Note that

$$\omega(G) \leq \Delta(G) + 1 \leq \bar{\omega}_k(G)$$

and observe that we can easily check whether $\omega(G) = \Delta(G) + 1$.

Hence, it is NP-hard to check whether $\bar{\omega}_k(G) = \Delta(G) + 1$.

Corollary

Let k be a fixed integer, $k \geq 2$. Unless $P = NP$, there cannot be a polynomial time algorithm that finds a k-club of size greater than $\Delta(G) + 1$ whenever such a k-club exists in the graph.

Some references


B. Balasundaram, S. Butenko, and I. Hicks. Clique relaxations in social network analysis: the maximum k-plex problem. *Operations Research*, 2011.


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


J. Pattillo, A. Veremyev, S. Butenko, V. Boginski. On the maximum quasi-clique problem. *Discrete Applied Math*, 2013


V. Boginski, S. Butenko, O. Shirokikh, S. Trukhanov, and J. Gil-Lafuente. A network-based data mining approach to portfolio selection via weighted clique relaxations. *Annals of Operations Research*, 2014.


A. Buchanan, J. S. Sung, V. Boginski, S. Butenko. On connected dominating sets of restricted diameter. *European Jour. of Oper. Res.*, 236: 410–418, 2014.


J. Pattillo, Y. Wang, and S. Butenko. Approximating 2-cliques in unit disk graphs. *Discrete Applied Math*, 2014.


S. Shahinpour and S. Butenko.Algorithms for the maximum k-club problem in graphs.*J. of Combinatorial Optim*, 2013.


A. Veremyev, O. A. Prokopyev, S. Butenko, and E. L. Pasiliao. Exact MIP-based approaches for finding maximum quasi-cliques and dense subgraphs. *Computational Optimization and Applications*, 2017.


C. Balasubramaniam and S. Butenko. On robust clusters of minimum cardinality in networks. *Annals of Operations Research*, 2017.


O. Yezerska, S. Butenko, and V. L. Boginski. Detecting robust cliques in graphs subject to uncertain edge failures. *Annals of Operations Research*, 2018.


O. Yezerska, F. Mahdavi Pajouh, A. Veremyev, and S. Butenko. Exact algorithms for the minimum s-club partitioning problem. *Annals of Operations Research*, 2018.

Outline

Continuous approaches

- Continuous formulations of discrete optimization problems provide an alternative viewpoint and may lead to new insights.
- New optimality conditions and bounds on the optimal values may result from such formulations.
- The continuous formulations of discrete problems provide opportunities for establishing fundamental complexity results for continuous optimization.
- Discrete optimization may benefit from advances in global optimization algorithms for solving general or special classes of nonconvex problems.
- On the other hand, continuous formulations of discrete optimization problems stimulate the development of global optimization algorithms.

Continuous approaches

DIMACS

Center for Discrete Mathematics and Theoretical Computer Science
Founded as a National Science Foundation Science and Technology Center

HOME | ABOUT | NEWS | PROGRAMS | EVENTS | GIVING

HOME > PROGRAMS > THEMED PROGRAMS > SPECIAL FOCUS ON BRIDGING CONTINUOUS AND DISCRETE OPTIMIZATION

Special Focus on Bridging Continuous and Discrete Optimization

Running 2018-2020

The DIMACS Special Focus on Bridging Continuous and Discrete Optimization is part of the DIMACS/Simons Collaboration on Bridging Continuous and Discrete Optimization, a Research Coordination Network led by DIMACS and the Simons Institute for the Theory of Computing to advance both the foundations and practical capabilities of optimization algorithms and methods.

Optimization capabilities touch our everyday lives through more efficient supply chains, better traffic management, more secure power grids, and a host of other important applications. In the short history of the field of mathematical optimization, advances in underlying theory, practical implementation, and raw computing power have brought us from solving linear programs with a few hundred variables to those with more than a million. Widely available general-purpose solvers make sophisticated tools for linear, integer, and nonlinear programming broadly accessible to practitioners. New applications, particularly those stemming from machine learning and data science, are now challenging the field with issues related to uncertainty, scale, speed, and complexity. The field is meeting these challenges with innovative new methods improving performance guarantees that had stood for decades. Many of these innovations bring together ideas from both continuous and discrete optimization.

Historically, continuous and discrete optimization have followed largely distinct trajectories and drawn inspiration from different branches of mathematics. The study of discrete optimization is most closely associated with discrete mathematics and theoretical computer science, while continuous optimization is rooted in the well-

Probabilistic method

- A feasible solution of a discrete optimization problem (P) usually consists of a finite set of elements (e.g., vertices or edges of a graph) satisfying some property, and the objective is often to maximize/minimize the size of this set.
- Let $f^*(P)$ denote the optimal objective value of (P).
- In probabilistic method, with each such element i we associate its probability x_i of being included (randomly and independently) in some feasible solution.

Probabilistic method

- Assume that (P) is a maximization problem.
- The expected size $f(x)$ of the set of chosen elements forming a feasible solution of (P) satisfies

$$f^*(P) \geq f(x)\ \forall x \in [0,1]^n \ \Rightarrow\ f^*(P) \geq \max_{x \in [0,1]^n} f(x).$$

- If there is $x^* \in [0,1]^n$ such that $f(x^*) \geq f^*(P)$, then

$$f^*(P) \leq f(x^*) \leq \max_{x \in [0,1]^n} f(x).$$

- So, we obtain a continuous formulation of (P):

$$f^*(P) = \max_{x \in [0,1]^n} f(x).$$

IUC number, $\alpha^{\omega}(G)$

- Pick, randomly and independently, each vertex i of V with probability x_i.
- Let U be the random set of picked vertices that do not form an open triangle with other picked vertices.
- The probability that a vertex v will be included in U is

$$x_v \prod_{u,w:\{u,v,w\}\in\Lambda(G)} (1 - x_u x_w).$$

- The expected size of U is given by

$$g(x) = \sum_{v\in V} x_v \prod_{u,w:\{u,v,w\}\in\Lambda(G)} (1 - x_u x_w),$$

IUC number, $\alpha^{\omega}(G)$

- Since the IUC number cannot be less than the expected size of U for any choice of the vector of probabilities x, we have

$$\alpha^{\omega}(G) \geq \max_{x \in [0,1]^{|V|}} g(x).$$

- On the other hand, noting that the characteristic vector x^* of any maximum IUC gives $g(x^*) = \alpha^{\omega}(G)$, we conclude that

$$\max_{x \in [0,1]^{|V|}} g(x) \geq g(x^*) \geq \alpha^{\omega}(G),$$

so, $\alpha^{\omega}(G) = \max_{x \in [0,1]^{|V|}} g(x)$.

IUC number, $\alpha^{\omega}(G)$

Proposition

$$\alpha^{\omega}(G) = \max_{x \in [0,1]^{|V|}} \sum_{v \in V} x_v \prod_{u,w:\{u,v,w\} \in \Lambda(G)} (1 - x_u x_w).$$

- Setting $x_1 = \ldots = x_n = y$, we obtain:

$$\alpha^{\omega}(G) \geq \max_{y \in [0,1]} \left(y \sum_{i \in V} \left(1 - y^2\right)^{\Lambda_i} \right) \geq \max_{y \in [0,1]} \left(ny \left(1 - y^2\right)^{\Lambda_{\max}} \right)$$

- Setting $y = 1/\sqrt{2\Lambda_{\max} + 1}$, we obtain the following bound:

$$\alpha^{\omega}(G) \geq \frac{2n\Lambda_{\max}}{(2\Lambda_{\max} + 1)^{3/2}}$$

Formulating a discrete problem as a nonlinear program

- An IP formulation of a discrete optimization problem can be converted into a continuous nonlinear formulation by replacing the binary constraint

$$x \in \{0, 1\}$$

with a continuous nonconvex equality constraint

$$x(1 - x) = 0$$

or with inequality constraints

$$x(1 - x) \leq 0, \quad 0 \leq x \leq 1.$$

Formulating a discrete problem as a nonlinear program

- Shor (1989) formulated the maximum independent set problem as

$$\alpha(G) = \max \sum_{i=1}^{n} x_i$$

subject to

$$x_i x_j = 0, \ (i,j) \in E;$$

$$x_i^2 - x_i = 0, \ i = 1, \ldots, n.$$

- Shor has shown that the optimal solution of its Lagrangian dual yields Lovász $\vartheta(G)$.

Formulating a discrete problem as a nonlinear program

- If we have

$$\max_{x \in \{0,1\}^n} f(x),$$

where $f(x)$ is linear with respect to each variable, then this problem is equivalent to

$$\max_{x \in [0,1]^n} f(x).$$

- For the maximum independent set problem we have

$$\alpha(G) = \max_{x \in [0,1]^n} \sum_{i \in V(G)} x_i - \sum_{(i,j) \in E(G)} x_i x_j.$$

Optimality conditions for binary QP

Let $X = \mathrm{diag}(x);\ e = [1, \ldots, 1]^T \in \mathbf{R}^n$.

Theorem (Beck and Teboulle, 2000)

For the problem $\min\left\{\frac{1}{2}x^T Q x + b^T x : x = \pm 1\right\}$ *any global minimizer* $x = \pm 1$ *satisfies*

$$\mathrm{diag(Q)}e \geq XQXe + Xb$$

and any $x = \pm 1$ *satisfying*

$$\lambda_{min}(Q)e \geq XQXe + Xb$$

is a global optimal solution.

Cubic formulation of the maximum IUC problem

Proposition

The maximum IUC problem can be formulated as follows:

$$\alpha^{\omega}(G) = \max_{x \in [0,1]^{|V|}} \left(\sum_{v \in V} x_v - \sum_{\{u,v,w\} \in \Lambda(G)} x_u x_v x_w \right)$$

Corollary

The IUC number $\alpha^{\omega}(G)$ satisfies the following inequality:

$$\alpha^{\omega}(G) \geq \begin{cases} \sqrt{\frac{4|V|^3}{27|\Lambda(G)|}}, & \text{if } |\Lambda(G)| \geq |V|/3, \\ |V| - |\Lambda(G)|, & \text{if } |\Lambda(G)| < |V|/3. \end{cases}$$

Quadratic formulation for the MEWC problem

Proposition

The MEWC problem can be formulated as follows:

$$\max_{\mathbf{x}\in[0,1]^n} \quad \frac{1}{2}\,\mathbf{x}^T\mathbf{Q}\,\mathbf{x}, \tag{P}$$

where,

$$\mathbf{Q}(i,j) = \begin{cases} 0, & i = j \\ w_{ij}, & \{i,j\} \in E \\ -\bar{w}_{ij}, & \{i,j\} \notin E, \end{cases}$$

and,

$$\bar{w}_{ij} = \max\{\sum_{k\in N(i)} w_{ik}, \sum_{k\in N(j)} w_{jk}\} + \epsilon \quad \forall\, \{i,j\} \notin E, i \neq j,$$

for an arbitrarily small $\epsilon > 0$.

Quadratic formulations for the clique number, $\omega(G)$

- Motzkin & Straus (1965):

$$1 - \frac{1}{\omega(G)} = \max\{x^T A_G x : e^T x = 1, x \geq 0\}.$$

- Regularization by Bomze et al. (1997):

$$1 - \frac{1}{2\omega(G)} = \max\left\{x^T \left(A_G + \frac{1}{2} I_n\right) x : e^T x = 1, x \geq 0\right\}.$$

Extension of Motzkin-Straus formulation to s-defective clique

- Let $A_{\bar{G}}(y)$ be the adjacency matrix $A_{\bar{G}}$ of $\bar{G}$, where each nonzero entry is replaced with the corresponding y_{ij}.
- Consider the following optimization problem.

$$\bar{f}_s(G) = \max_{(x,y)\in\bar{\Omega}_s} \bar{f}^s_G(x,y),$$

where

$$\bar{f}^s_G(x,y) = x^T(A_G + A_{\bar{G}}(y))x,$$

$$\bar{\Omega}_s = \{(x,y) \in \mathbb{R}^n_+ \times \{0,1\}^{\bar{m}} : e_n^T x = 1, e_{\bar{m}}^T y \leq s\}.$$

Proposition

Let $\bar{\omega}_s(G)$ denote the s-defective clique number of G, then

$$1 - \frac{1}{\bar{\omega}_s(G)} = \bar{f}_s(G).$$

Extension of Turán's theorem to s-defective clique

- In 1941, Turán established the following relation between the number of edges m, the number of vertices n and the clique number $\omega(G)$ of any simple graph G:

$$m \leq \left(1 - \frac{1}{\omega(G)}\right) \frac{n^2}{2}.$$

- Satisfied at equality if and only if G is a complete $\omega(G)$-partite graph with balanced parts.

Extension of Turán's theorem to s-defective clique

- In 1941, Turán established the following relation between the number of edges m, the number of vertices n and the clique number $\omega(G)$ of any simple graph G:

$$m \leq \left(1 - \frac{1}{\omega(G)}\right) \frac{n^2}{2}.$$

- Satisfied at equality if and only if G is a complete $\omega(G)$-partite graph with balanced parts.

Theorem (Turán's theorem for s-defective clique)

Assume that $\omega_s(G) < n$ (that is, V is not an s-defective clique). Then

$$m \leq \left(1 - \frac{1}{\bar{\omega}_s(G)}\right) \frac{n^2}{2} - s.$$

Extension of Motzkin-Straus formulation to s-plex

Consider the following optimization problem.

$$f_s(G) = \max_{(x,y)\in\Omega_s} f_G^s(x,y),$$

where

$$f_G^s(x,y) = x^T(A_G + A_{\bar{G}}(y))x,$$

$$\Omega_s = \{(x,y) \in \mathbb{R}_+^n \times \{0,1\}^{\bar{m}} : e_n^T x = 1, I_{\bar{G}} y \leq (s-1)e_n\},$$

$I_{\bar{G}}$ is the $n \times \bar{m}$ incidence matrix of the complement graph $\bar{G}$.

Proposition

Let $\omega_s(G)$ denote the s-plex clique number of G, then

$$1 - \frac{1}{\omega_s(G)} = f_s(G).$$

Extension of Turán's theorem to s-plex

- Given a simple graph $G = (V, E)$ and a positive integer k, $M_k \subseteq E$ is called a k-matching in G if the subgraph induced by M_k does not have a vertex of degree $> k$.
- The maximum k-matching problem, which is to find a k-matching with the largest number of edges in G (denoted by $\mu_k(G)$), is polynomial time solvable.

Extension of Turán's theorem to s-plex

- Given a simple graph $G = (V, E)$ and a positive integer k, $M_k \subseteq E$ is called a k-matching in G if the subgraph induced by M_k does not have a vertex of degree $> k$.
- The maximum k-matching problem, which is to find a k-matching with the largest number of edges in G (denoted by $\mu_k(G)$), is polynomial time solvable.

Theorem (Turán's theorem for s-plex)

$$m \leq \left(1 - \frac{1}{\omega_s(G)}\right)\frac{n^2}{2} - \mu_{s-1}(\bar{G}).$$

V. Stozhkov, A. Buchanan, S. Butenko, and V. Boginski. Generalizing Turan's theorem to clique relaxations. Working paper.

Fractional formulations for $\alpha(G)$

Formulation I:

$$\alpha(G) = \max_{x \in [0,1]^n} \sum_{i \in V(G)} \frac{x_i}{1 + \sum_{j \in N(i)} x_j}$$

Formulation II:

$$\alpha(G) = \max_{x \in [0,1]^n} \sum_{i \in V(G)} \frac{x_i}{1 + \sum_{j \in N(i)} x_j} - \sum_{(i,j) \in E, i<j} x_i x_j$$

B. Balasundaram and S. Butenko. On a polynomial fractional formulation for independence number of a graph. *Journal of Global Optimization*, 2006.

Fractional formulations for $\alpha(G)$

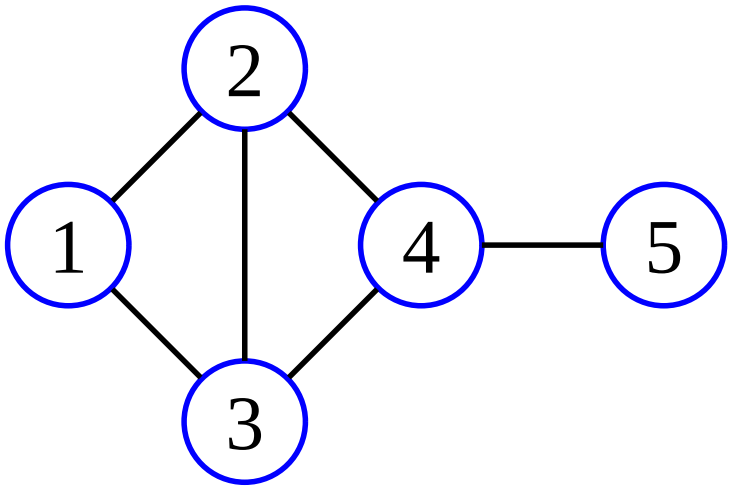
- Wei's lower bound (Wei, 1981) is obtained trivially by setting all variables to 1:

$$\alpha(G) \geq \sum_{i \in V(G)} \frac{1}{1 + d_i}$$

- Every local maximum point is a binary vector (both formulations).
- $x^* \equiv I^*$ denotes the correspondence of a binary vector to a subset of vertices $I^* = \{i \in V : x_i^* = 1\}$.

Global maxima of Formulation I

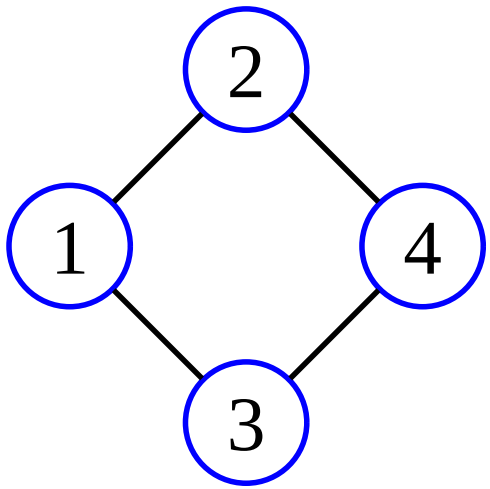
- If x^* is a global maximizer and $x^* \equiv I^*$, then I^* induces an *independent union of cliques* (IUC).
- Example: $\alpha(G) = 2$



The optimum is also achieved at $\{1,2,3,5\}$, which is an IUC.

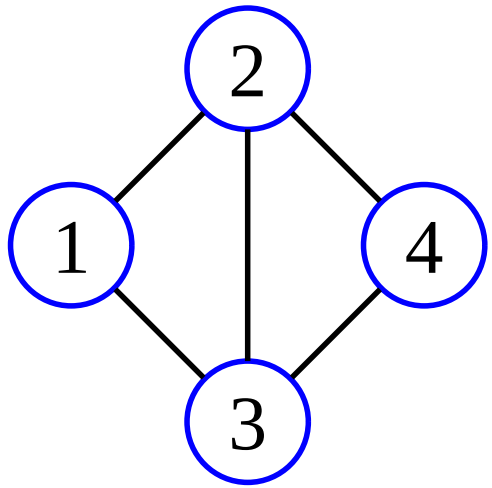
Local maxima of Formulation I

- Even though a global maximizer always corresponds to an IUC, a local maximizer may not have the same property.
- The sets $\{1, 4\}$ and $\{2, 3\}$ are maximal, as well as maximum independent sets. But the point $x = [1, 1, 1, 1]^T \equiv V$ is also a local maximum.



Local maxima of Formulation I

If x^0 is a point of local maximum and $x^0 \equiv I^0$, then I^0 is a *dominating set*. Converse is NOT true!



$I = \{1, 2, 3\}$ is a dominating set and a *maximal IUC*. But $x = [1, 1, 1, 0]^T \equiv I$ is not a local maximum.

Outline

Recognizing generalized convexity

- A graph is called well-covered if every independent set is contained in a maximum independent set of G, i.e., every maximal independent set is maximum in a well-covered graph.
- Recognizing a well-covered graph is NP-hard.
- Using Bomze's regularization of the Motzkin-Straus formulation for the maximum clique problem, we obtain the following result:

Proposition

It is NP-hard to check if a non-convex quadratic program over a standard simplex has the property that every local minimum is a global minimum.

Duality gap recognition

Theorem (Busygin & Pasechnik, 2003)

For a graph G it is NP-hard to decide whether there is a gap between its independence and clique partition numbers

$$\bar{\chi}(G) - \alpha(G) > 0$$

provided some minimum clique partition of G is given.

Proposition

Recognizing whether the duality gap is positive (i.e., checking whether the gap between the optimal value of a nonlinear optimization problem and its Lagrangian dual is nonzero) is NP-hard for quadratic problems.

Outline

Algorithms based on continuous approaches

- Exact algorithms
  - Non-convex polynomial-time solvable relaxations can be used instead of convex relaxations
- Approximation algorithms
  - Can we prove tight approximation bounds based on non-convex relaxations?
- Heuristics
  - How to extract high-quality solutions for combinatorial problem using continuous formulations?
  - Combine different formulations (e.g., variable objective search)

A Construction heuristic algorithm

$$\max_{\mathbf{x}\in[0,1]^n} \quad \frac{1}{2}\,\mathbf{x}^T\mathbf{Q}\,\mathbf{x}, \tag{P}$$

- **(P)** is "hard" to solve.

A Construction heuristic algorithm

$$\max_{\mathbf{x}\in[0,1]^n} \quad \frac{1}{2}\,\mathbf{x}^T\mathbf{Q}\,\mathbf{x}, \tag{P}$$

- **(P)** is "hard" to solve.
- We propose a **construction heuristic** based on solving a surrogate of **(P)**:

$$\max_{\mathbf{x}^T\mathbf{x}=1} \quad \frac{1}{2}\,\mathbf{x}^T\mathbf{Q}\,\mathbf{x} \tag{P'}$$

  Stationary points of **(P')** are normalized eigenvectors of $\mathbf{Q}$.

- For each eigenvector of $\mathbf{Q}$, we:
  - treat it as an approximate characteristic vector (of a set of vertices)
  - extract the corresponding clique from the graph
  - keep (return) the heaviest clique

A combinatorial branch-and-bound algorithm

Algorithm 1 Branch-and-bound procedure

function BRANCH(G, L, C, C^*, W, W^*) ▷ L: candidate list
  while $L \neq \{\}$ **do**
    $p =$ PRUNE(G, L, C, W, W^*)
    if $p =$ `true` **then**
      return
    else
      $v \leftarrow$ last vertex in L
      $\delta W = \sum_{i \in C} w_{iv}$; $C \leftarrow C \cup \{v\}$; $W \leftarrow W + \delta W$;
      $L_v =$ an array of neighbors of v in L
      if $L_v \neq \{\}$ **then**
        BRANCH(G, L_v, C, C^*, W, W^*)
      else if $W > W^*$ **then**
        $C^* \leftarrow C$; $W^* \leftarrow W$
      end if
      $C \leftarrow C \backslash \{v\}$; $W \leftarrow W - \delta W$
    end if
    $L \leftarrow L \backslash \{v\}$
  end while
  return
end function

Quadratic relaxation upper bound

- Consider $G' = G[C \cup L]$,

$$W^*_{G'} = \max_{\mathbf{x} \in [0,1]^{n'}} \frac{1}{2} \mathbf{x}^T \mathbf{Q}' \mathbf{x}$$

Quadratic relaxation upper bound

- Consider $G' = G[C \cup L]$,

$$W^*_{G'} = \max_{\mathbf{x} \in [0,1]^{n'}} \frac{1}{2} \mathbf{x}^T \mathbf{Q}' \mathbf{x}$$

- Given $x_i = 1, \forall i \in C$, (define: $q_j = \sum_{i \in C} w_{ij}, \forall j \in L$)

$$W^*_{G'} = W(C) + \max_{\mathbf{x} \in \{0,1\}^{|L|}} \left(\mathbf{q}^T \mathbf{x} + \frac{1}{2} \mathbf{x}^T \mathbf{Q}'_L \mathbf{x} \right)$$

Quadratic relaxation upper bound

- Consider $G' = G[C \cup L]$,

$$W^*_{G'} = \max_{\mathbf{x} \in [0,1]^{n'}} \frac{1}{2}\,\mathbf{x}^T \mathbf{Q}'\,\mathbf{x}$$

- Given $x_i = 1, \forall i \in C$, (define: $q_j = \sum_{i \in C} w_{ij}, \forall j \in L$)

$$W^*_{G'} = W(C) + \max_{\mathbf{x} \in \{0,1\}^{|L|}} \left(\mathbf{q}^T \mathbf{x} + \frac{1}{2}\,\mathbf{x}^T \mathbf{Q}'_L\,\mathbf{x} \right)$$

- $\mathbf{b} = \frac{1}{2}\,\mathbf{1}_{|L|}$

$$W^*_{G'} \leq W(C) + \left(\begin{array}{ll} \mathcal{Z}_L = \max\limits_{\mathbf{x} \in \mathbb{R}^{|L|}} & \mathbf{q}^T \mathbf{x} + \frac{1}{2}\,\mathbf{x}^T \mathbf{Q}'_L\,\mathbf{x} \\ s.t. & (\mathbf{x} - \mathbf{b})^T (\mathbf{x} - \mathbf{b}) = \frac{|L|}{4} \end{array} \right)$$

Results

Table: Solution time ("$>$" $\equiv$ "$> 10,800\ sec.$")

		CPU (sec.)			
Name	W^*	CBQ	Gouveia & Martins		IP_{BASE}
brock200-1	21,230	**3,047.565**	$>$		$>$
brock200-2	6,542	**7.436**	9,464.240	(F1)	$>$
brock200-3	10,303	**55.905**	$>$		$>$
brock200-4	13,967	**188.031**	$>$		$>$
C125.9	66,248	**4,558.170**	$>$		$>$
C250.9	-	$>$	$>$		$>$
c-fat200-1	7,734	**0.483**	3.870	(F61)	31.296
c-fat200-2	26,389	**0.890**	33.260	(F2)	49.671
c-fat200-5	168,200	$>$	155.300	(F1)	**134.578**
hamming6-2	32,736	4.437	**0.300**	(F11)	17.000
hamming6-4	396	**0.031**	1.970	(F1)	6.468
hamming8-2	-	$>$	$>$		$>$
hamming8-4	12,360	**439.437**	$>$		$>$
johnson16-2-4	3,808	**84.687**	$>$		$>$
johnson8-4-4	6,552	**0.687**	2.340	(F11)	65.171
keller4	6,745	**42.218**	$>$		$>$
p-hat300-1	3,321	**3.281**	1,273.050	(F2)	8,489.750
p-hat300-2	31,564	**171.281**	$>$		$>$
p-hat300-3	-	$>$	$>$		$>$

Quality of the quadratic relaxation bound

Name	#QR	#SUM	avg. QR/SUM
brock200-1	38,819,232 (99.82 %)	71,511 (0.18 %)	0.66
brock200-2	108,456 (99.82 %)	200 (0.18 %)	0.57
brock200-3	884,461 (99.82 %)	1,590 (0.18 %)	0.61
brock200-4	2,711,652 (99.85 %)	4,071 (0.15 %)	0.62
C125.9	26,539,316 (99.95 %)	13,835 (0.05 %)	0.72
c-fat200-1	5 (1.35 %)	366 (98.65 %)	1.27
c-fat200-2	0 (0.00 %)	4,861 (100.00 %)	1.30
hamming6-2	11,836 (36.68 %)	20,431 (63.32 %)	1.02
hamming6-4	790 (80.28 %)	194 (19.72 %)	0.70
hamming8-4	4,990,643 (97.13 %)	147,215 (2.87 %)	0.65
johnson16-2-4	3,076,874 (99.49 %)	15,649 (0.51 %)	0.40
johnson8-4-4	10,863 (98.19 %)	200 (1.81 %)	0.55
keller4	704,846 (99.71 %)	2,061 (0.29 %)	0.44
p-hat300-1	22,217 (98.43 %)	354 (1.57 %)	0.58
p-hat300-2	1,694,985 (99.13 %)	14,820 (0.87 %)	0.74

S. Hosseinian, D. B. M. M. Fontes, and S. Butenko. A nonconvex quadratic optimization approach to the maximum edge weight clique problem. *Journal of Global Optimization*, to appear, DOI: 10.1007/s10898-018-0630-5.

Outline

Polynomial constraints representation

- Consider a closed set in the form

$$S = \{x \in \mathbb{R}^n : P_1(x) \leq 0, \ldots, P_m(x) \leq 0\},$$

  where m is a positive integer and $P_i,\ i = 1, \ldots, m$ are polynomial functions of the n variables $x_1, \ldots, x_n$.

- Scheiderer (1989) and Bröcker (1991) have shown that S can be represented by at most $\binom{n+1}{2}$ polynomial constraints, i.e., there exist polynomial functions $Q_1, \ldots, Q_{n(n+1)/2}$ such that

$$S = \{x \in \mathbb{R}^n : Q_1(x) \leq 0, \ldots, Q_{n(n+1)/2}(x) \leq 0\}.$$

Polynomial constraints representation

Bosse, Grötschel and Henk (2005) prove further extensions of the above results dealing with polytopes.

- They show explicitly how every n-dimensional polytope $P = \{x \in \mathbb{R}^n : Ax \leq b\}$, where $A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$, can be described by at most $2n - 1$ polynomial inequalities.

Question: *Can we represent P using only a small number of low-degree polynomials?*


thank you
danke
merci
gracias
obrigado
dziękuję
bedankt
teşekkür ederim
go raibh maith agat
mochchakkeram
ngiyabonga
謝謝
спасибо
рахмат
Баярлалаа
dank je
tapadh leat
sukriya
kop khun krap
terima kasih
감사합니다
ευχαριστώ
arigatô
dakujem
grazie
takk
taiku
sagolun
hvala
mauruuru
dankie
kiitos
faafetai lava
vinaka
спасибі
blagodaram
misaotra
matondo
paldies
grazzi
mahalo
barka
mersi
welalin
tack
spas
kia ora
akun
dankon
aciu
köszönöm
nandri
nanni
dhanyavad
bayarlalaa
grâcie
enkosi
хвала
asante
manana
obrigada
tenki
murakoze
chokrane
djiere dieuf
tau
дякую
mamnun
chnorakaloutioun
gratias ago
gràcies
sulpáy
sobodi
dēkuji
mèsi
didi madloba
kam sah hamnida
najis tuke
rahmat
ありがとう
tanemirt
rahmet
diolch
dhanyavadagalu
shukriya
merce
мерси
trugarez
তোমাকে ধন্যবাদ
xièxie