Nested Clustering on a Graph

Dave Morton
Industrial Engineering & Management Sciences
Northwestern University

Joint work with Gökçe Kahvecioğlu and Mike Nehme

Clustering on a Graph

Optimal attack and reinforcement of a network
W.H. Cunningham (1985)

Clustering on a Graph

- Given $G = (V, E)$. Each edge has cost $c_e > 0$, $e \in E$
- Delete edges $K \subset E$ to form $G' = (V, E \setminus K)$
- Cost: $c(K) = \sum_{e \in K} c_e$

Clustering on a Graph

- Given $G = (V, E)$. Each edge has cost $c_e > 0$, $e \in E$
- Delete edges $K \subset E$ to form $G' = (V, E \setminus K)$
- Cost: $c(K) = \sum_{e \in K} c_e$
- Gain: $g(K) =$ number of connected components of $G' = (V, E \setminus K)$
  - Let $r(K)$ be the rank of $G' = (V, E \setminus K)$, where rank is the largest number of edges that can participate in a forest
  - Then $g(K) = |V| - r(K)$

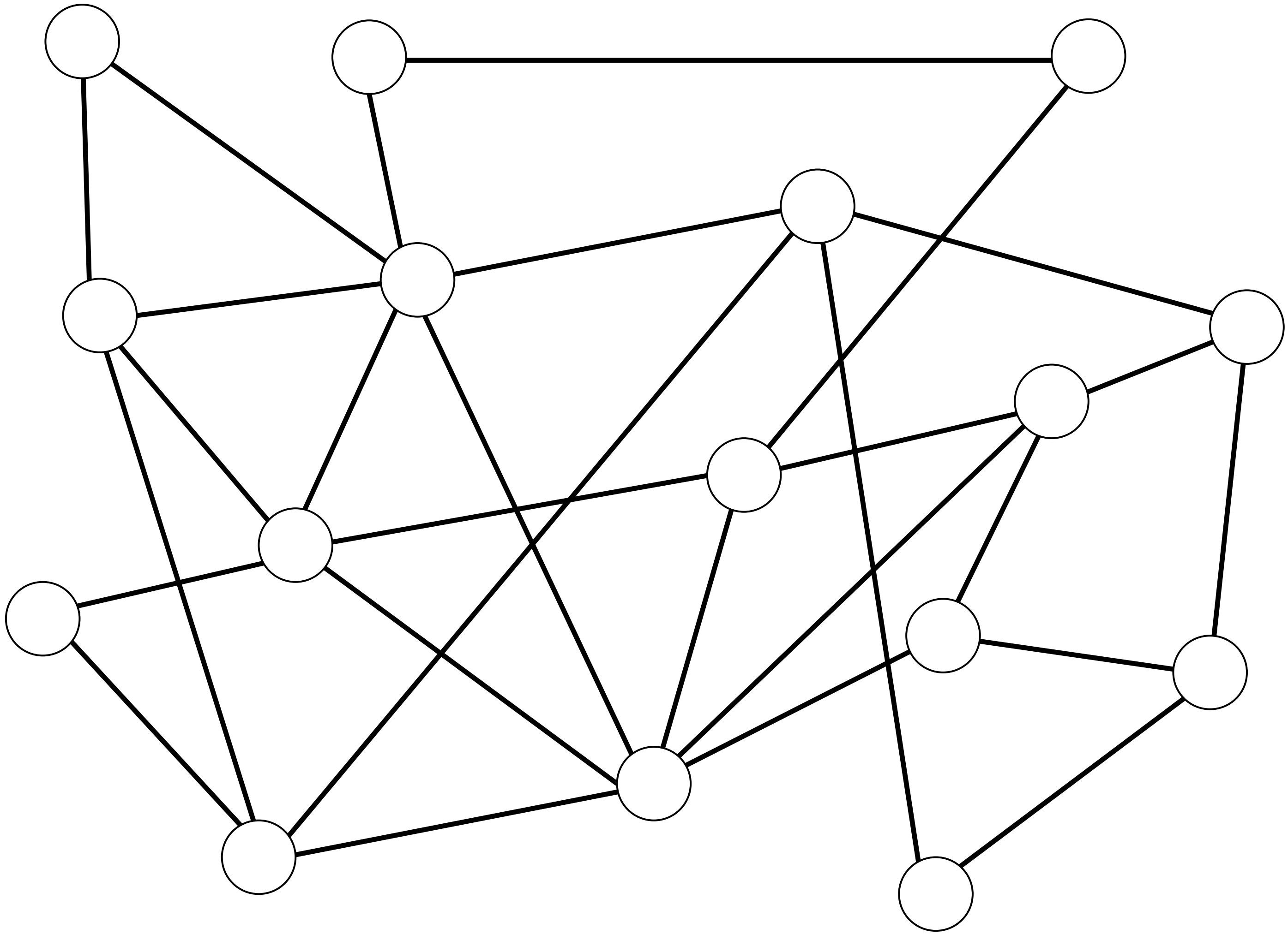
Clustering on a Graph

- Model:

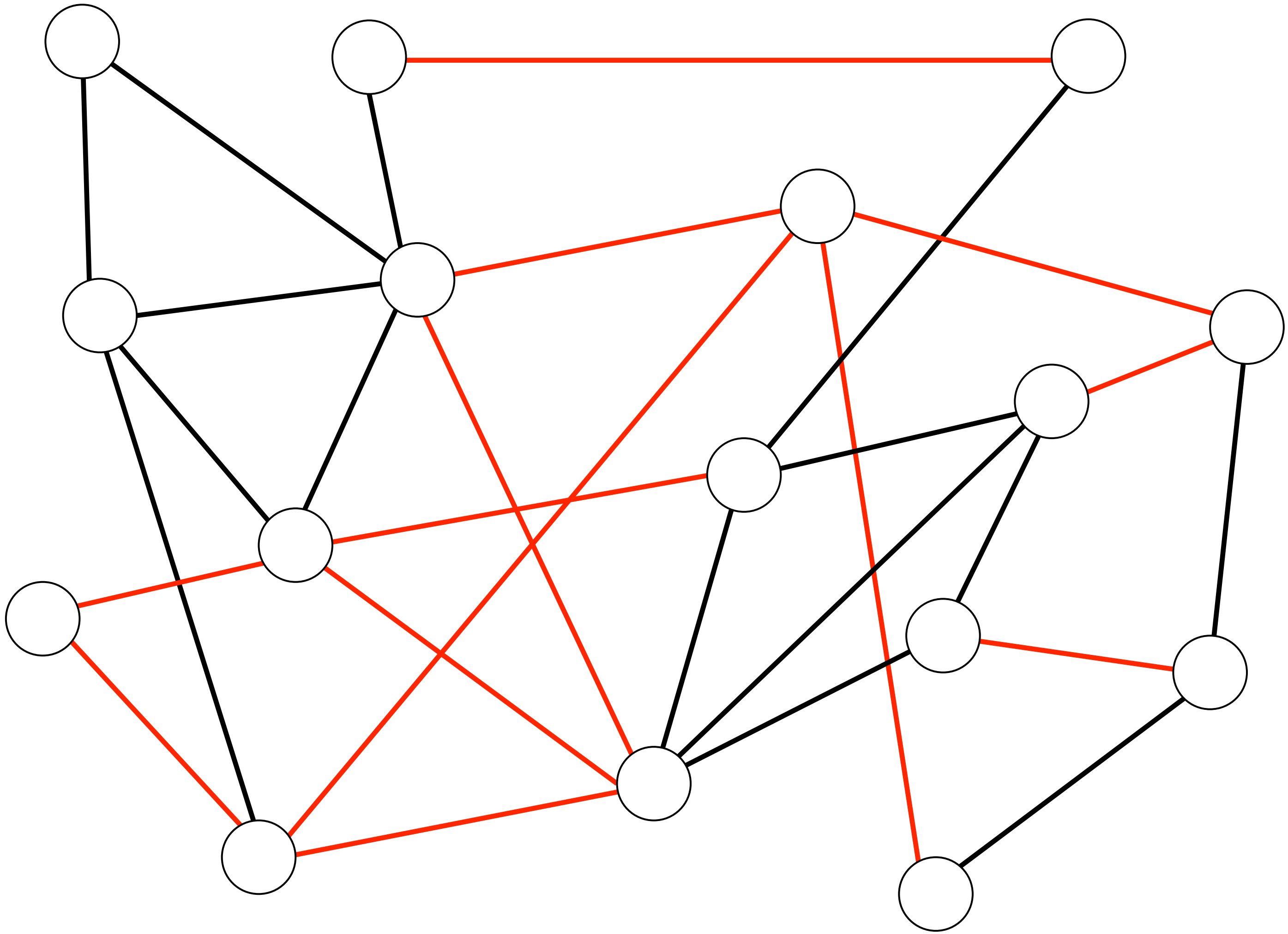
$$\begin{aligned} \max_{K \subset E} \quad & g(K) \\ \text{s.t.} \quad & c(K) \leq b \end{aligned}$$

- If $c(K) = |K|$: Partition graph into as many pieces as possible, subject to cardinality constraint on number of edges we delete

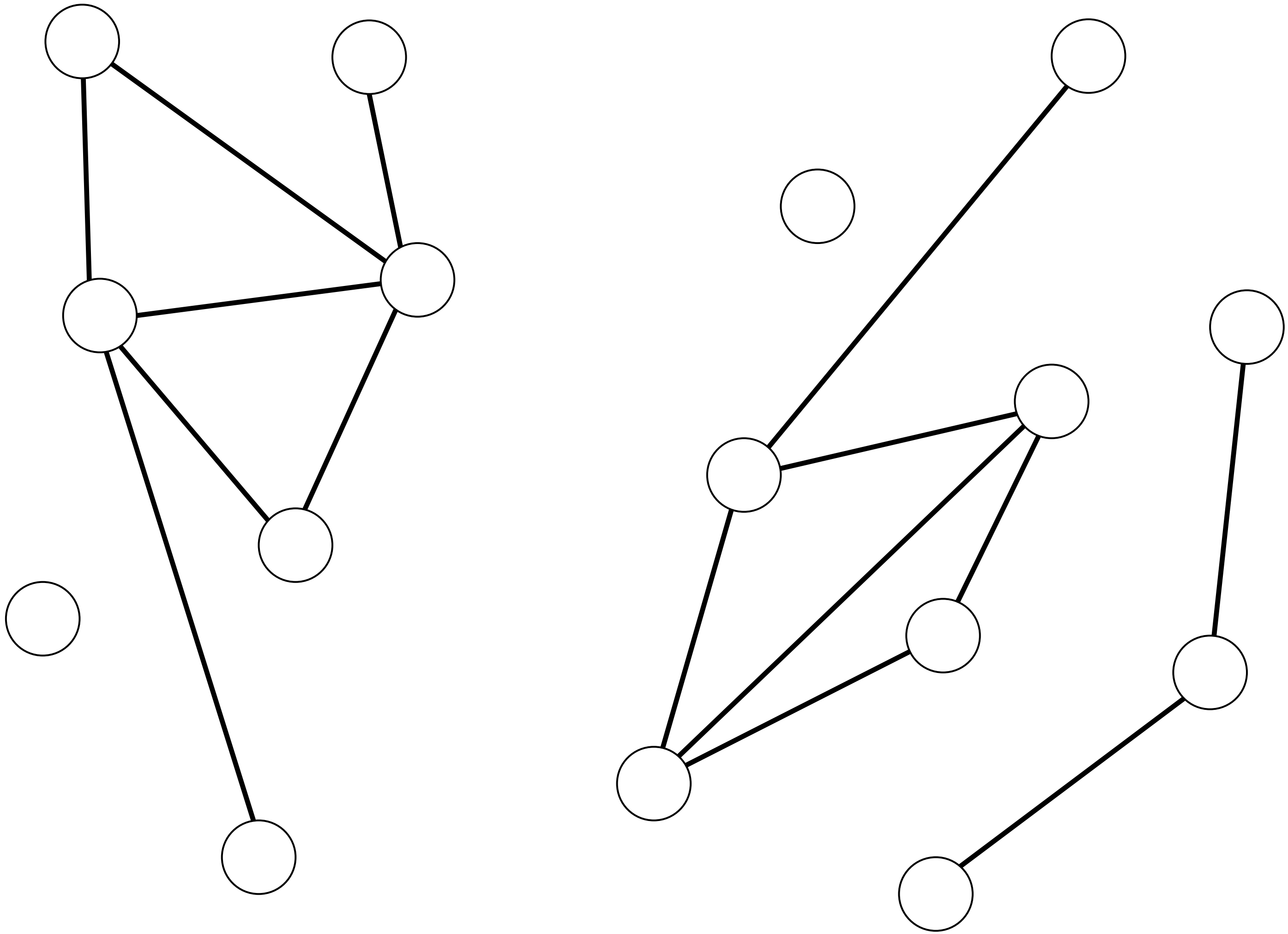
Clustering on a Graph



Clustering on a Graph



Clustering on a Graph



Clustering on a Graph

- A related model:
$$\max_{K \subset E} \quad g(K) - \lambda c(K),$$
where $\lambda > 0$ is given

- Easier model and important for reasons we'll see shortly

- Cunningham's strength of a graph:
$$\min_{K \subset E} c(K)/[g(K) - 1]$$

- Bicriteria view: Find Pareto efficient solutions, maximizing $g(K)$ and minimizing $c(K)$

- $g(K)$ is a supermodular function

Maximize a supermodular function subject to a submodular knapsack constraint

A Bicriteria Combinatorial Optimization Problem

- Let S be a finite universal set
- Let $g : 2^S \to \mathbb{R}$ be a supermodular gain function
- Let $c : 2^S \to \mathbb{R}$ be an increasing, submodular cost function
- Model:

$$\begin{aligned} \max_{K \subset S} \quad & g(K) \\ \text{s.t.} \quad & c(K) \leq b \end{aligned} \tag{1}$$

- Bicriteria view: Find Pareto efficient solutions, maximizing $g(K)$ and minimizing $c(K)$
- Nestedness: Let K_b and $K_{b'}$ solve model (1) for b and b', $b < b'$. These optimal solutions are nested, if $K_b \subset K_{b'}$

Super- and Submodular Functions

- $g : 2^S \to \mathbb{R}$ is a supermodular function, provided

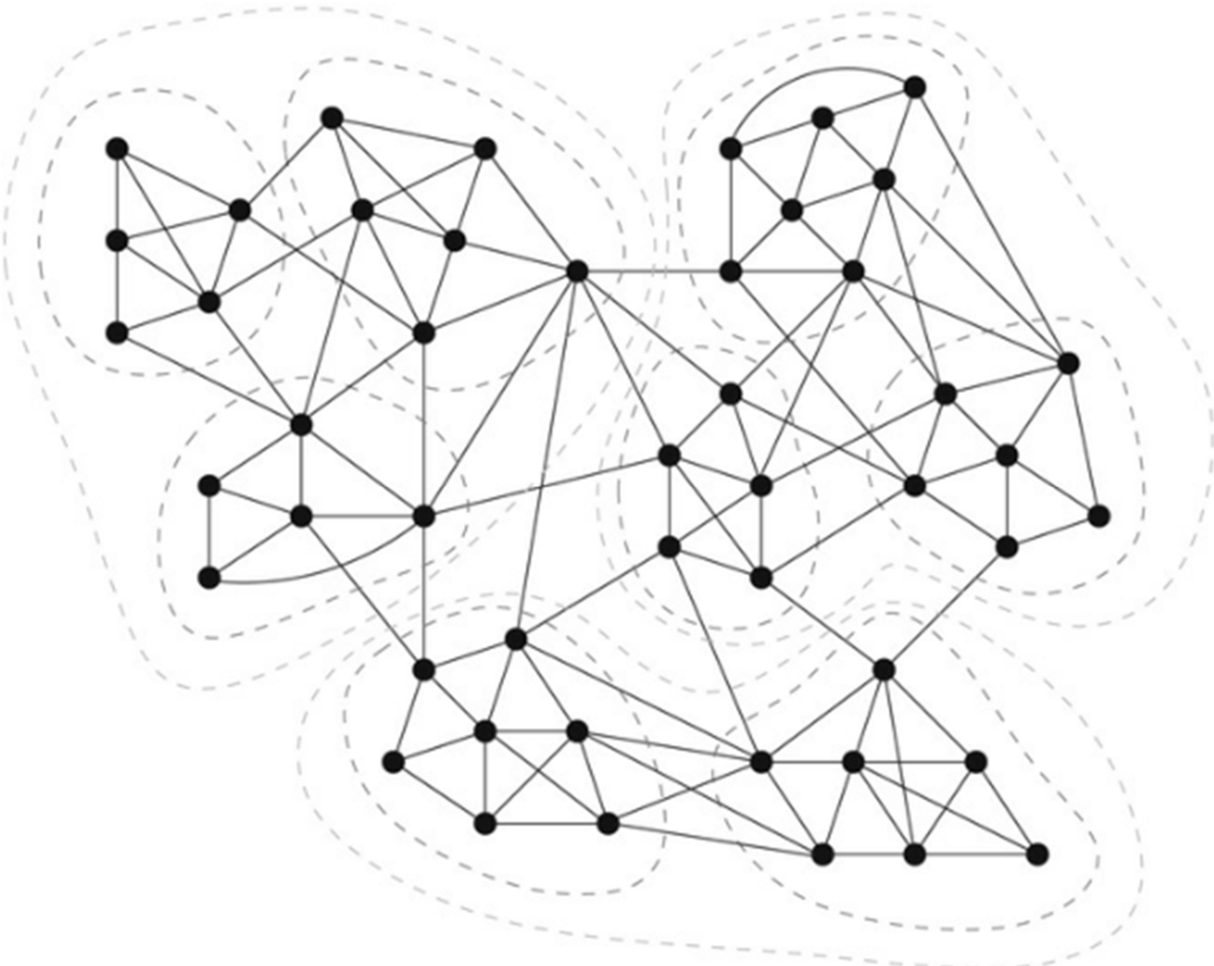
$$g(B \cup \{k\}) - g(B) \geq g(A \cup \{k\}) - g(A)$$

where $A \subset B \subset S$ and where $k \in S \setminus B$

- $c : 2^S \to \mathbb{R}$ is submodular if $-c(\cdot)$ is supermodular

- A function is modular if it is both super- and submodular

Nested Clustering on a Graph



Geometry and Nestedness under Supermodularity

- Model:

$$\begin{aligned} \max_{K \subset S} \quad & g(K) \\ \text{s.t.} \quad & c(K) \leq b \end{aligned} \tag{1}$$

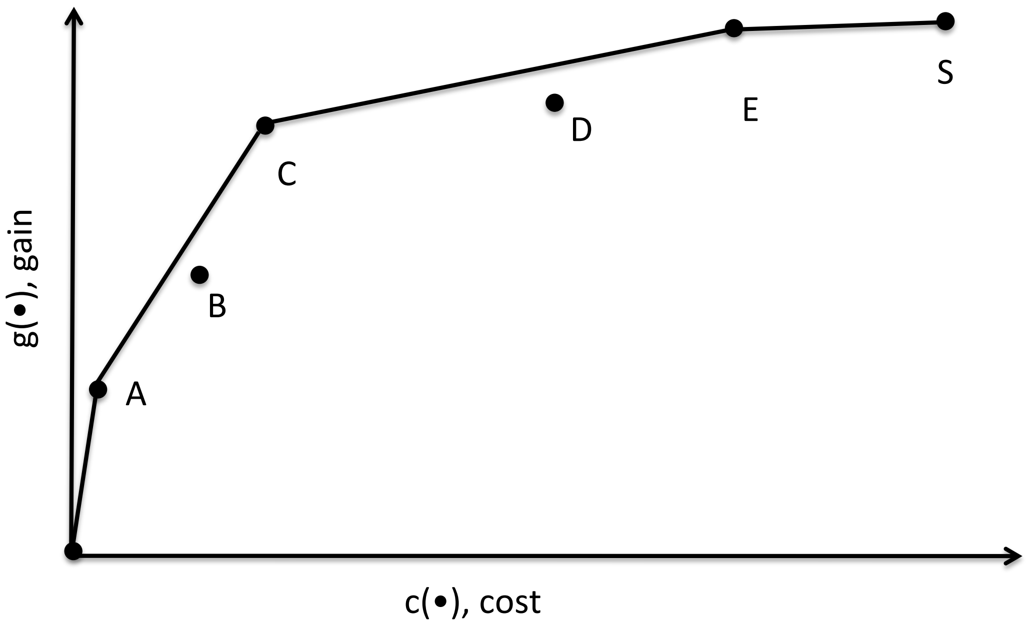
- Assume $c(\cdot)$ is submodular and increasing. And $g(\cdot)$ is supermodular

- Let $A, B \subset S$ satisfy $c(A) < c(B)$.

  Gain-to-cost ratio: $m : 2^S \times 2^S \to \mathbb{R}$ is:

$$m(A, B) = \frac{g(B) - g(A)}{c(B) - c(A)}$$

Gain-to-Cost Ratio



$$m(A, B) = \frac{g(B) - g(A)}{c(B) - c(A)}$$

Geometry and Nestedness under Supermodularity

Lemma 1 *Let* $B \subset S$ *be a solution of model (1) on the concave envelope of the efficient frontier. Then,*

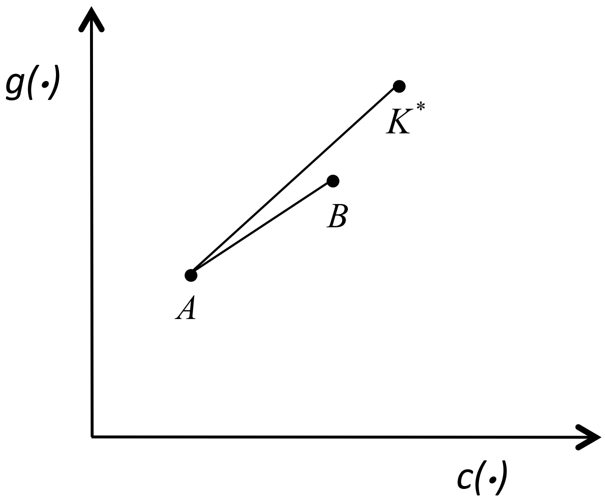
$$m(A,B) = \max_{K \subset S : c(K) \geq c(B)} m(A,K) \;\; \forall\, A : c(A) < c(B)$$

and

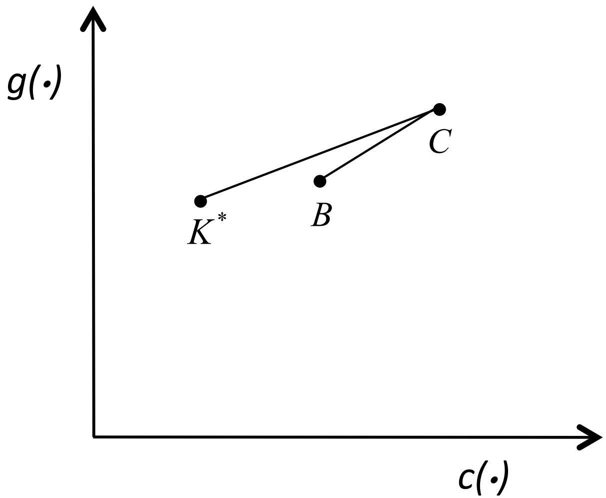
$$m(B,C) = \min_{K \subset S : c(K) \leq c(B)} m(K,C) \;\; \forall\, C : c(C) > c(B)$$

Geometry and Nestedness under Supermodularity

Lemma 1 (in pictures): *Let $B \subset S$ be a solution of model (1) on the concave envelope of the efficient frontier. Then the following is impossible; i.e., there is no such K^*:*



(a)



(b)

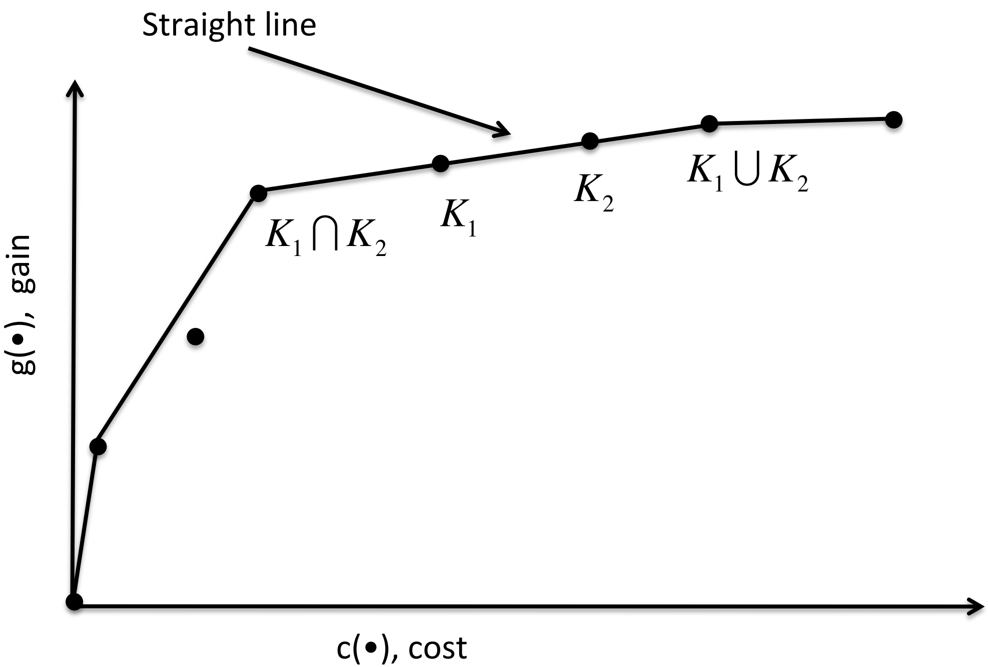
Geometry and Nestedness under Supermodularity

Lemma 2 *Assume $c(\cdot)$ is submodular and increasing and $g(\cdot)$ is supermodular. Let $K_1, K_2 \subset S$ be solutions on the concave envelope of the efficient frontier of model (1) with $K_1 \not\subset K_2$ and $K_2 \not\subset K_1$. Then*

$$m(K_1 \cap K_2, K_1) = m(K_2, K_1 \cup K_2) = m(K_1 \cap K_2, K_1 \cup K_2).$$

Geometry and Nestedness under Supermodularity

Lemma 2 (in pictures): Assume $c(\cdot)$ is submodular and increasing and $g(\cdot)$ is supermodular. Then



$$m(K_1 \cap K_2, K_1) = m(K_2, K_1 \cup K_2) = m(K_1 \cap K_2, K_1 \cup K_2)$$

Proof of Lemma 2

- $K_1 \cap K_2 \subset K_2$. So,

$$g(K_1) - g(K_1 \cap K_2) \leq g(K_1 \cup K_2) - g(K_2)$$

$$c(K_1) - c(K_1 \cap K_2) \geq c(K_1 \cup K_2) - c(K_2)$$

- Thus

$$m(K_1 \cap K_2, K_1) \leq m(K_2, K_1 \cup K_2) \tag{1}$$

- Applying Lemma 1 with $A = K_1 \cap K_2$ and $B = K_1$ yields:

$$m(K_1 \cap K_2, K_1 \cup K_2) \leq m(K_1 \cap K_2, K_1). \tag{2}$$

- Applying Lemma 1 with with $B = K_2$ and $C = K_1 \cup K_2$ yields:

$$m(K_2, K_1 \cup K_2) \leq m(K_1 \cap K_2, K_1 \cup K_2). \tag{3}$$

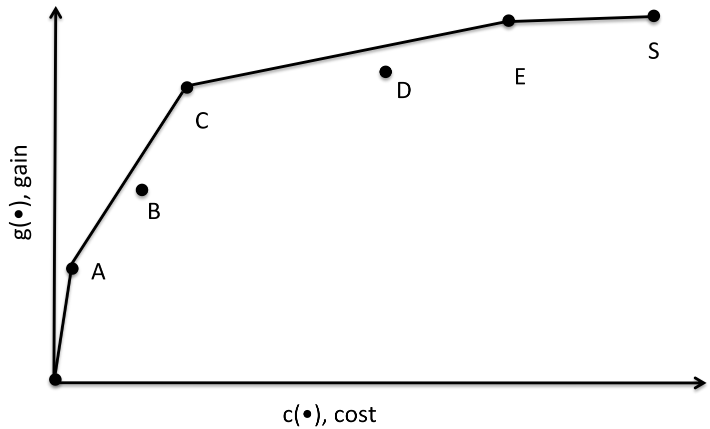
Taken together, inequalities (1)-(3) yield the desired result.

Geometry and Nestedness under Supermodularity

Theorem 3 *Assume* $c(\cdot)$ *is submodular and increasing and* $g(\cdot)$ *is supermodular. Let* $K_1, K_2 \subset S$ *be extreme points on the concave envelope of the efficient frontier of model (1). Then either* $K_1 \subset K_2$ *or* $K_2 \subset K_1$*. Moreover, if* $c(K_1) = c(K_2)$ *then* $K_1 = K_2$*.*

Geometry and Nestedness under Supermodularity

$$\max_{K \subset S} \quad g(K)$$
$$\text{s.t.} \quad c(K) \le b$$



- Assume $c(\cdot)$ is submodular and increasing and $g(\cdot)$ is supermodular
- Extreme points of concave envelope of efficient frontier are nested
- Obtain those solutions in strongly polynomial time via

$$\max_{K \subset S} \quad g(K) - \lambda c(K)$$

Okay. But, how do we solve the graph clustering problem?

$$\begin{aligned} \max_{K \subset S} \quad & g(K) \\ \text{s.t.} \quad & c(K) \leq b \end{aligned}$$

or

$$\max_{K \subset S} \quad g(K) - \lambda c(K)$$

LP for Minimum Spanning Tree

$$
\begin{aligned}
\min_{x} \quad & \sum_{e \in E} c_e x_e \\
\text{s.t.} \quad & \sum_{e \in E} x_e = |V| - 1 \\
& \sum_{\substack{e=(i,j) \in E \\ i,j \in S}} x_e \leq |S| - 1, S \subset V, S \neq \emptyset \\
& 0 \leq x_e \leq 1, e \in E.
\end{aligned}
$$

LP for Maximum Number of Edges in a Forest

$$
\begin{aligned}
r(E) = \max_x \quad & \sum_{e \in E} x_e \\
\text{s.t.} \quad & \sum_{\substack{e=(i,j)\in E \\ i,j \in S}} x_e \leq |S| - 1, S \subset V, S \neq \emptyset \\
& 0 \leq x_e \leq 1, e \in E,
\end{aligned}
$$

Recall:

- Let $r(K)$ be the rank of $G' = (V, E \setminus K)$, where rank is the largest number of edges that can participate in a forest
- Then $g(K) = |V| - r(K)$

LP for $g(K)$

$$
\begin{aligned}
g(K) = |V| - \max_x & \sum_{e \in E \setminus K} x_e \\
\text{s.t.} & \sum_{\substack{e=(i,j) \in E \setminus K \\ i,j \in S}} x_e \leq |S| - 1, S \subset V, S \neq \emptyset \\
& 0 \leq x_e \leq 1, e \in E \setminus K \\
= |V| + \min_x & \sum_{e \in E \setminus K} -x_e \\
\text{s.t.} & \sum_{\substack{e=(i,j) \in E \setminus K \\ i,j \in S}} x_e \leq |S| - 1, S \subset V, K \neq \emptyset \\
& 0 \leq x_e \leq 1, e \in E \setminus K
\end{aligned}
$$

LP for $g(y)$

Let $K = \{e : y_e = 1, e \in E\}$

$$
\begin{aligned}
g(y) = |V| + \min_x \quad & \sum_{e \in E} -x_e \\
\text{s.t.} \quad & \sum_{\substack{e=(i,j) \in E \\ i,j \in S}} x_e \le |S| - 1, S \subset V, S \ne \emptyset \\
& 0 \le x_e \le 1 - y_e, e \in E
\end{aligned}
$$

$$
\begin{aligned}
= |V| + \min_x \quad & \sum_{e \in E} (y_e - 1) x_e \\
\text{s.t.} \quad & \sum_{\substack{e=(i,j) \in E \\ i,j \in S}} x_e \le |S| - 1, S \subset V, S \ne \emptyset \ : \ \pi_S \\
& 0 \le x_e \le 1, e \in E \ : \ \gamma_e
\end{aligned}
$$

$$
\begin{aligned}
= |V| + \max_{\pi,\gamma} \quad & \sum_{S \subset V} (|S| - 1) \pi_S + \sum_{e \in E} \gamma_e \\
\text{s.t.} \quad & \sum_{S : i,j \in S} \pi_S + \gamma_e \le y_e - 1, e = (i,j) \in E \\
& \pi_S \le 0, S \subset V, S \ne \emptyset \\
& \gamma_e \le 0, e \in E.
\end{aligned}
$$

MIP for Knapsack-constrained Graph Clustering

A MIP for model (1) is then:

$$\begin{aligned}
\max_{y,\pi,\gamma} \quad & \sum_{S\subset V} (|S|-1)\pi_S + \sum_{e\in E} \gamma_e \\
\text{s.t.} \quad & \sum_{S:i,j\in S} \pi_S + \gamma_e \leq y_e - 1, e=(i,j)\in E \\
& \sum_{e\in E} c_e y_e \leq b \\
& \pi_S \leq 0, S\subset V, S\neq\emptyset \\
& \gamma_e \leq 0, e\in E \\
& y_e \in \{0,1\}, e\in E
\end{aligned}$$

Pricing problem for column generation is well-known max-flow problem on an auxiliary graph with $|V|+2$ nodes, just like in MST problem.

No, really. How do we solve the graph clustering problem?

$$\max_{K \subset S} \quad g(K) - \lambda c(K)$$

Solving Sequence of Max-Flow Problems Solves Graph Clustering Problem

1. Cunningham (1985) solves $|E|$ max-flow problems on a graph with $|V|+2$ nodes
2. Barahona (1992) solves at most $|V|$ max-flow problems on a graph with $|V|+2$ nodes
3. Baïou, Barahona and Mahjoub (2000) solve at most $|V|$ max-flow problems on a graph with $|k|+2$ nodes at iteration k
4. Preissmann and Sebó (2008) solve $|V|$ max-flow problems on a graph with at most $|k|+2$ nodes at iteration k

Max-flow problems are the same as in the MST problem.

How do we solve the *nested* graph clustering problem?

$$\max_{K \subset S} \quad g(K) - \lambda c(K) \qquad \forall \lambda > 0$$

Solving Sequence of *Parametric* Max-Flow Problems Solves *Nested* Graph Clustering Problem

1. Cunningham (1985)
2. Barahona (1992)
3. Baïou, Barahona, and Mahjoub (2000)
4. Preissmann and Sebó (2008)

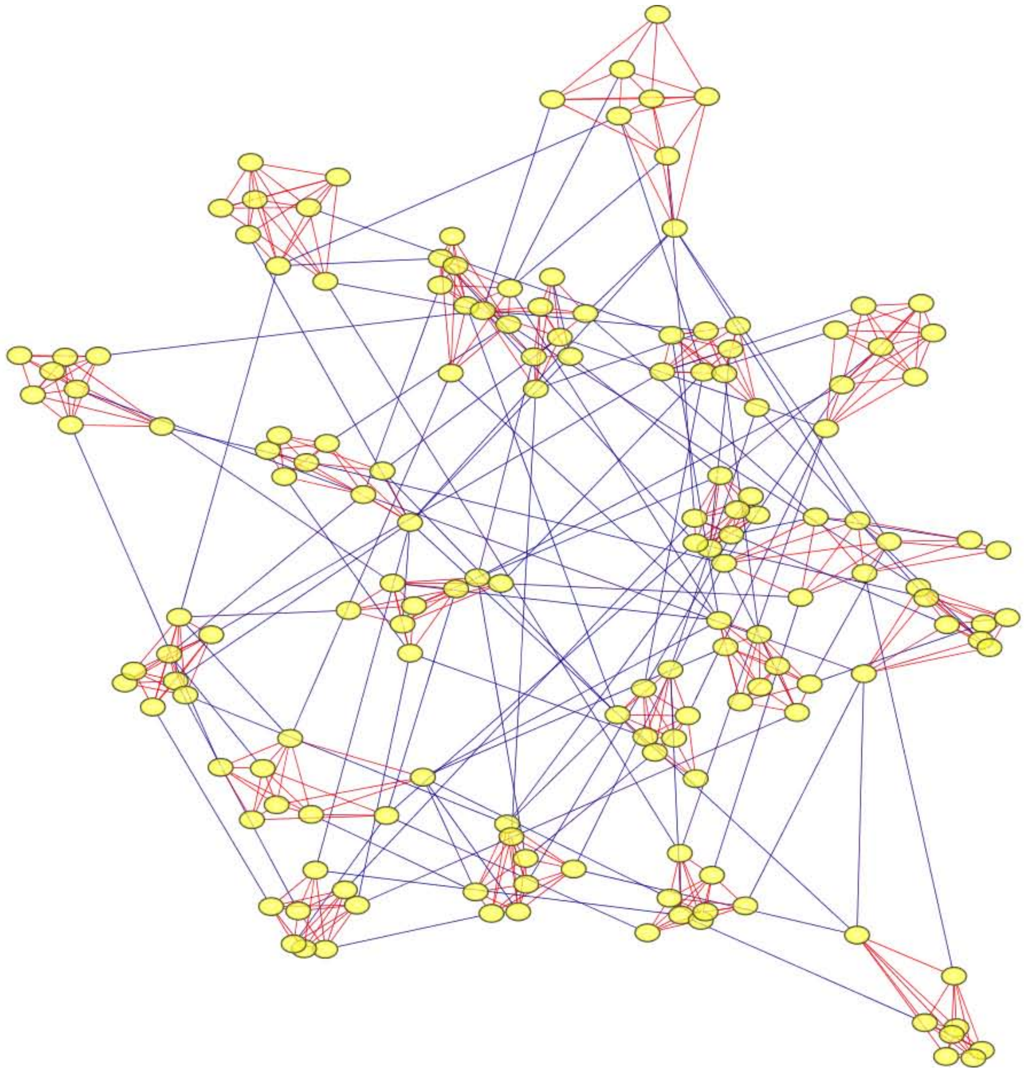
- Each algorithm works for fixed $\lambda > 0$
- We modify each, solving a parametric max-flow problem in λ
- This yields family of nested (hierarchical) clusters on the concave envelope of the efficient frontier

Parametric Max Flow

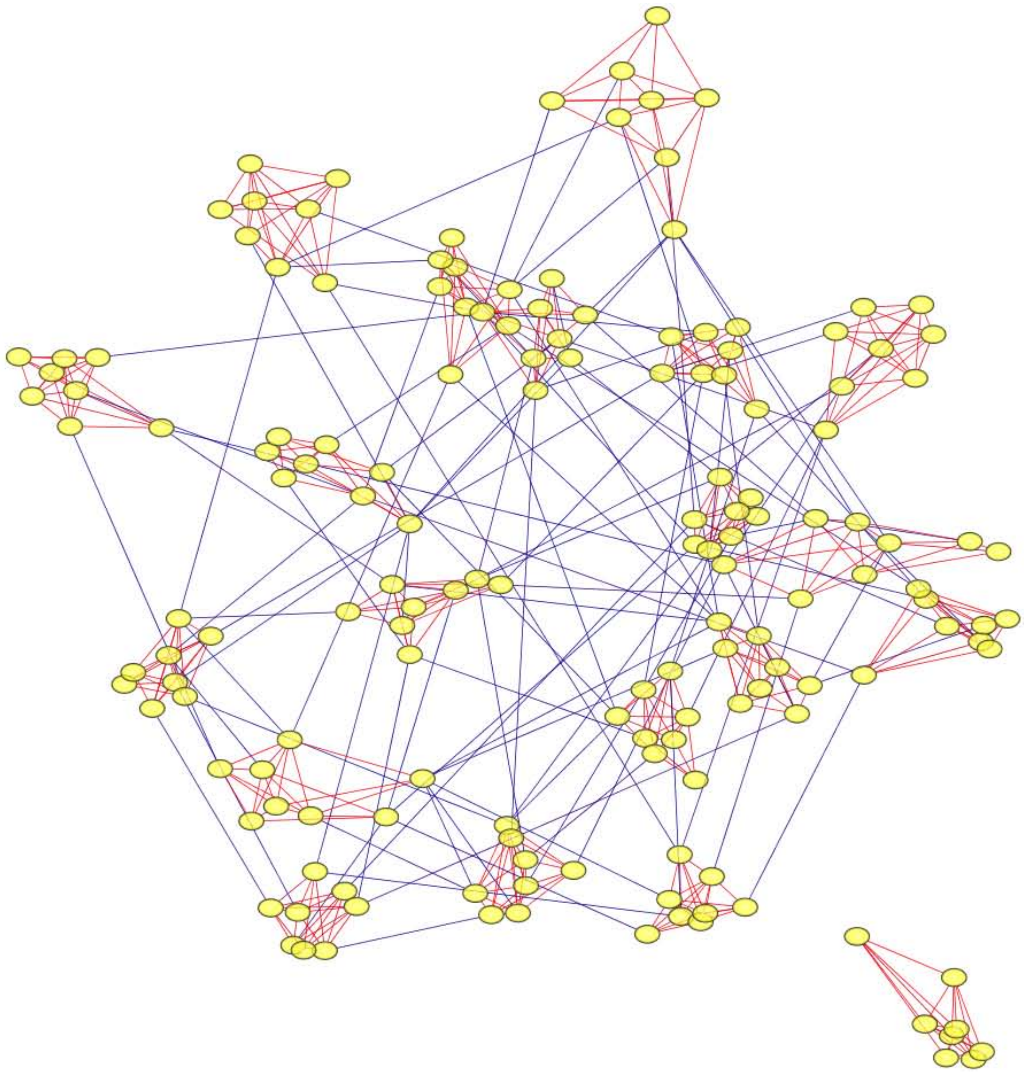
- In general, parametric LP and parametric max flow can have exponentially many break points
- But, we have nested property, and hence, at most $|V|$ break points
- Parametric push-relabel algorithm has same complexity as for fixed λ: Gallo, Grigoriadis and Tarjan (1989)
- Ditto for pseudo-flow algorithm (Hochbaum 2008) and others

We have preliminary implementation of Preissmann and Sebó (2008) with parametric max-flow in Python/Gurobi

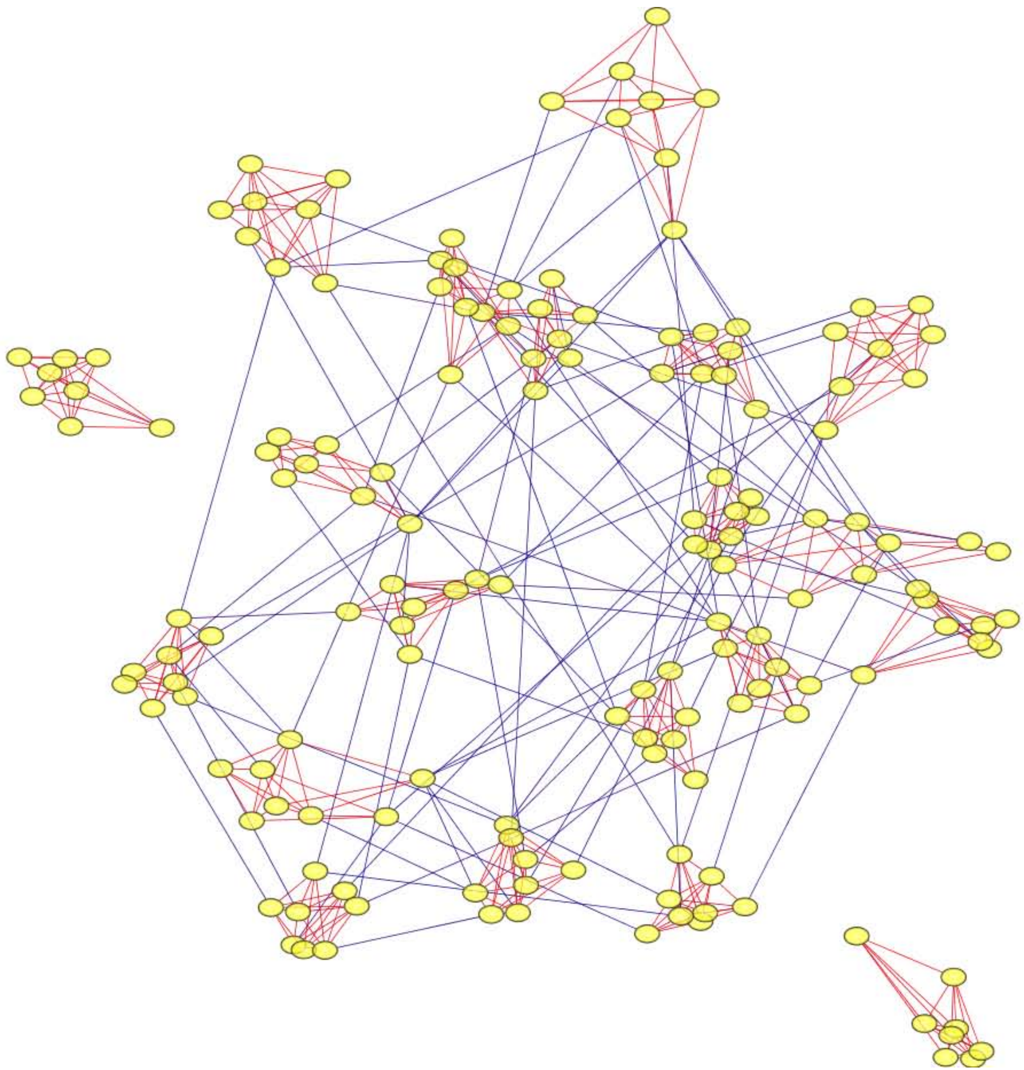
Relaxed Caveman Graph



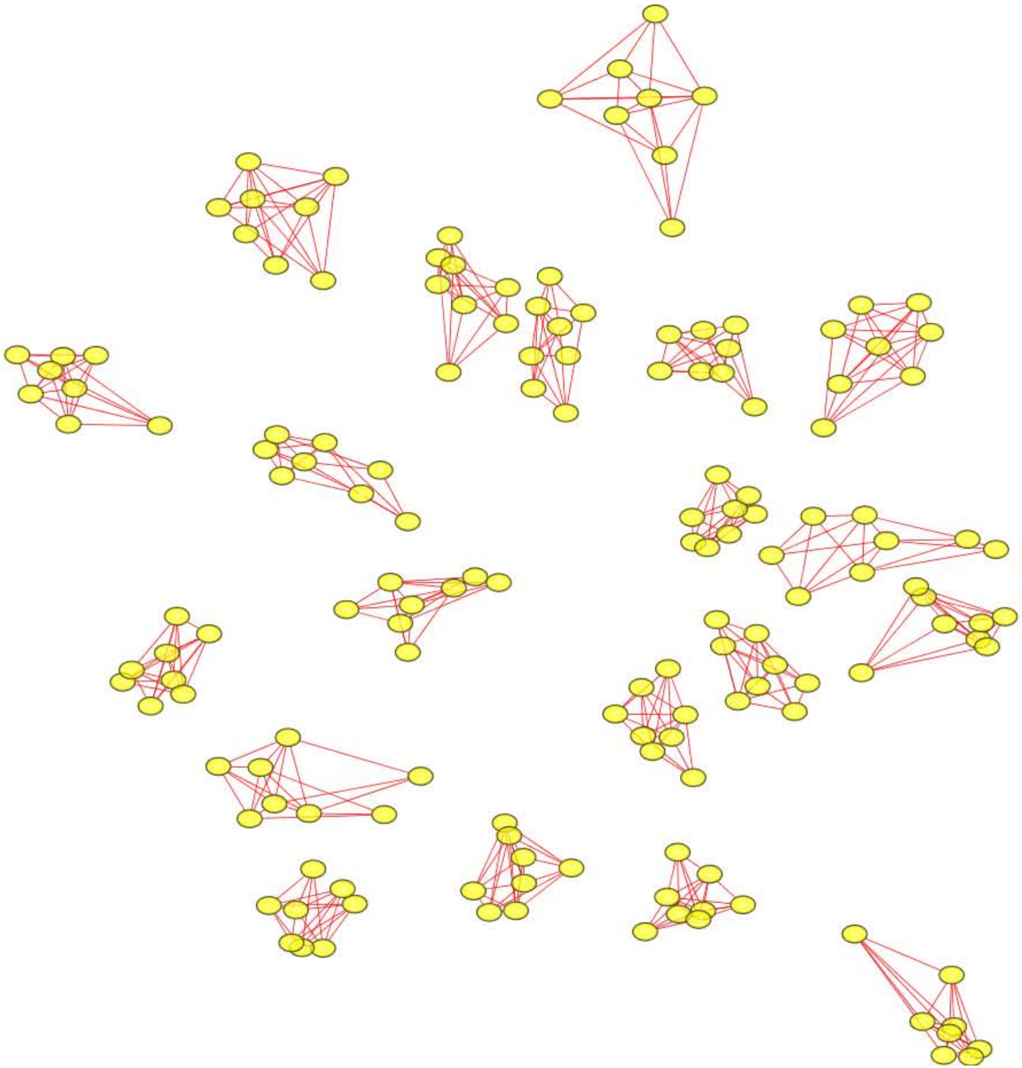
Relaxed Caveman Graph: $g(K) = 2$



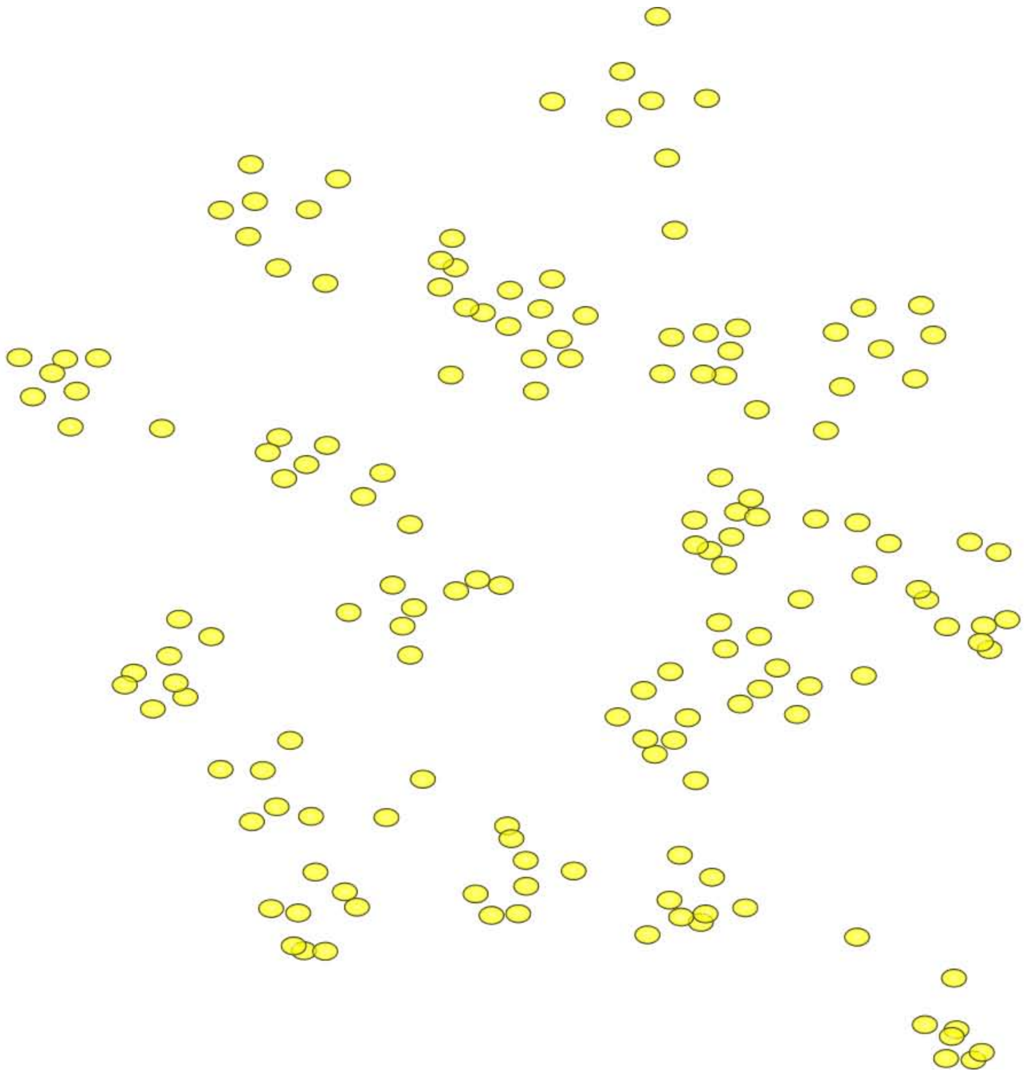
Relaxed Caveman Graph: $g(K) = 3$



Relaxed Caveman Graph: $g(K) = 20$



Relaxed Caveman Graph: $g(K) = 160$



Summary: Nested Clustering on a Graph

- Bicriteria model
  - maximize gain: number of clusters
  - minimize cost: weight of edges removed
- Gain is supermodular and cost is submodular, increasing
- Pareto efficient solutions on concave envelope of efficient frontier
  - computed in polynomial time
  - nested
- Proposed algorithm
  - combines Preissmann and Sebő (2008) and parametric max flow
  - solves nested clustering problem in same complexity as for fixed λ
- Value of, and connections to, MIP formulation?