Approximation Algorithms for the k-Set Packing Problem

Marek Cygan

Institute of Informatics
University of Warsaw

20th October 2016, Warszawa

Warm-up

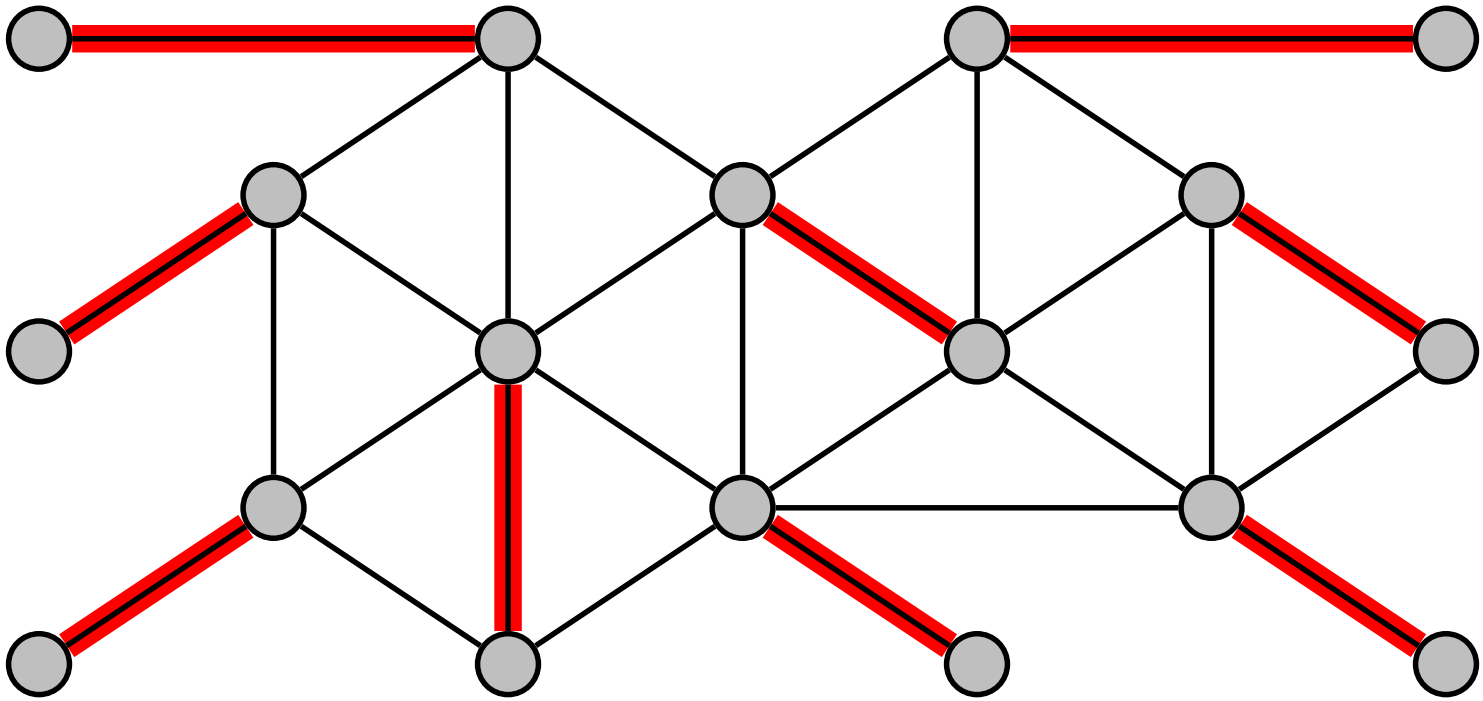
Warm-up (introduction to local search):

1. Maximum matching - toy example.
2. Local search for maximum matching.
3. Analysis.
4. Terminology.

Maximum Matching

Maximum Matching

Input: an undirected graph $G = (V, E)$.
Goal: find a maximum size vertex disjoint subset $M \subseteq E$.



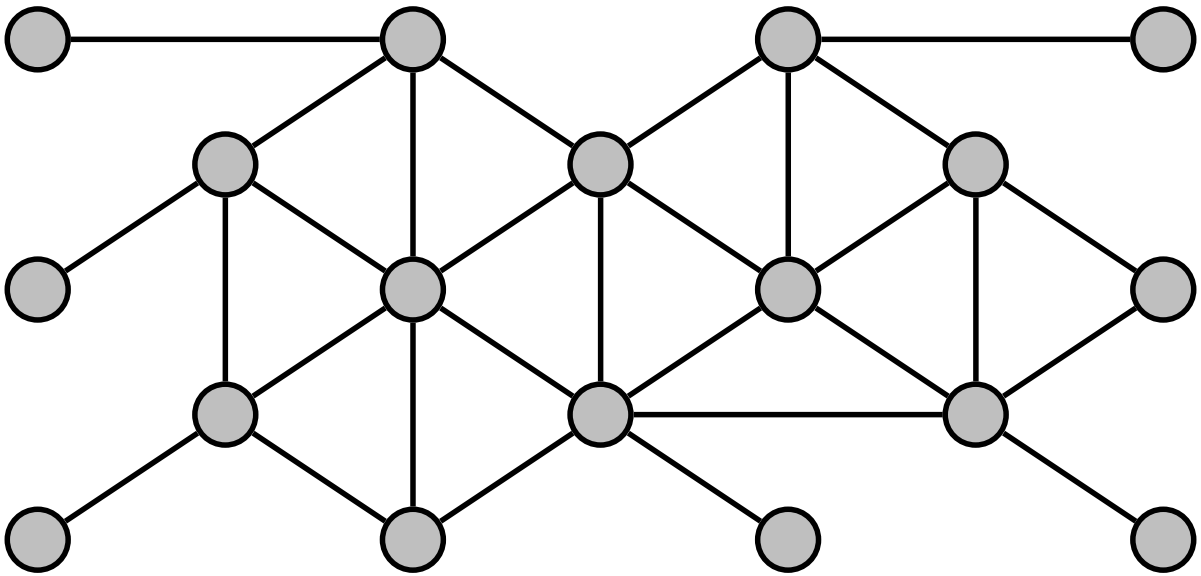
Maximum Matching

- Maximum matching can be found in polynomial time.
- As a toy example, consider the following routine.
- Initialize $M = \emptyset$.
- As long as possible improve M by applying one of the rules:

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$

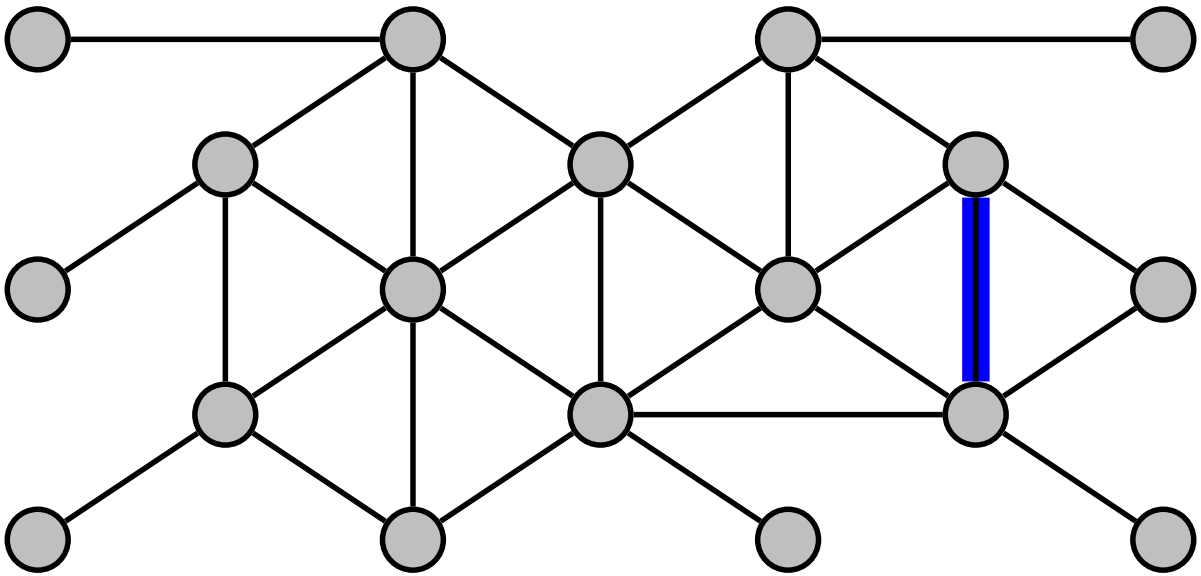
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



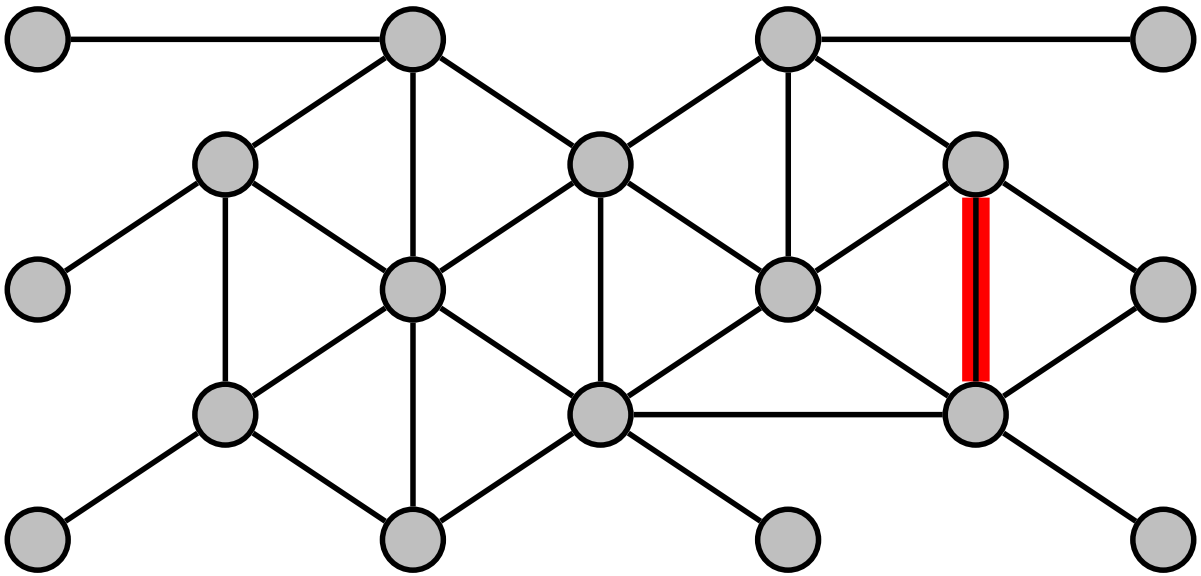
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



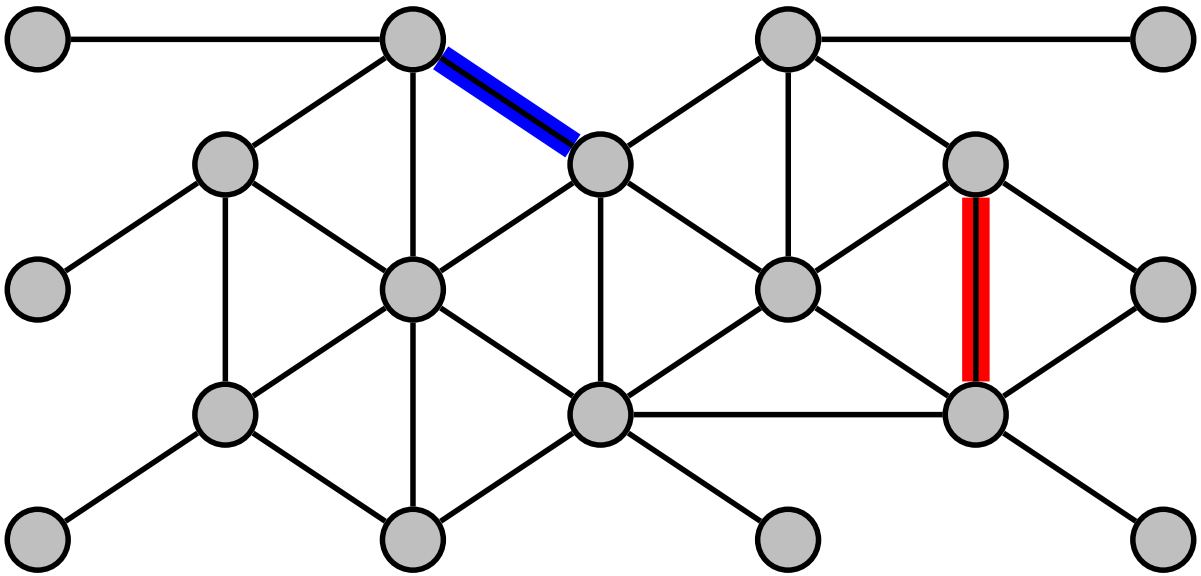
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



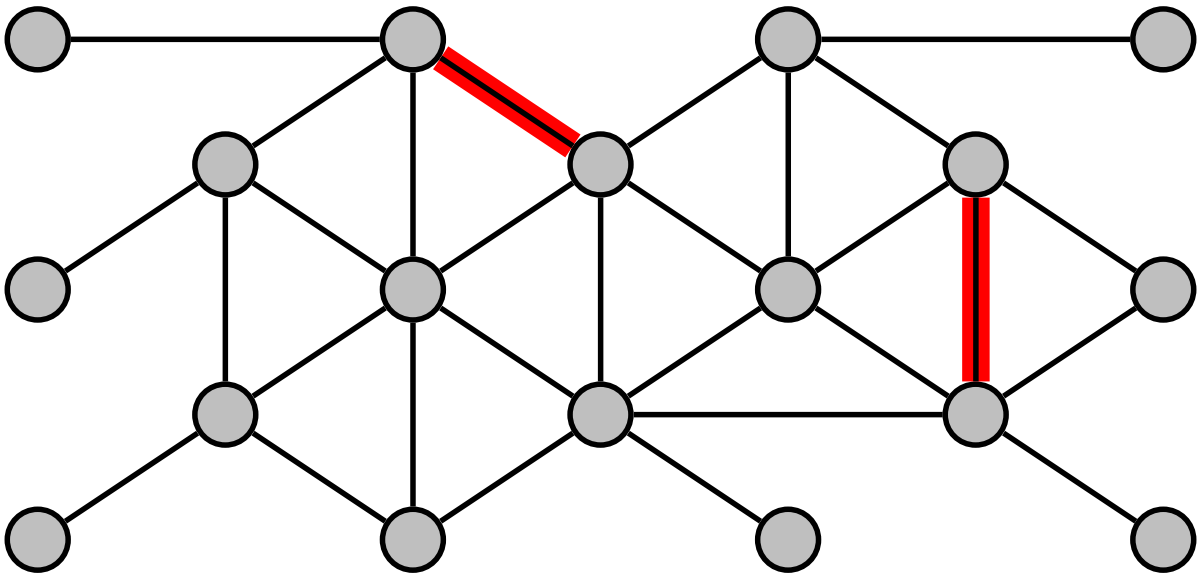
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



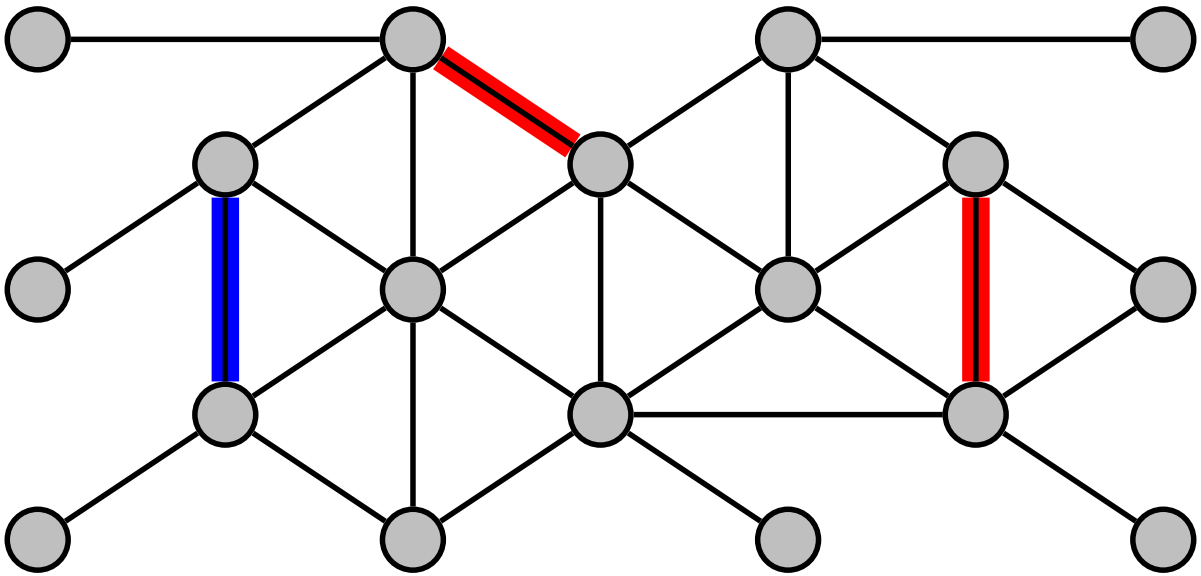
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



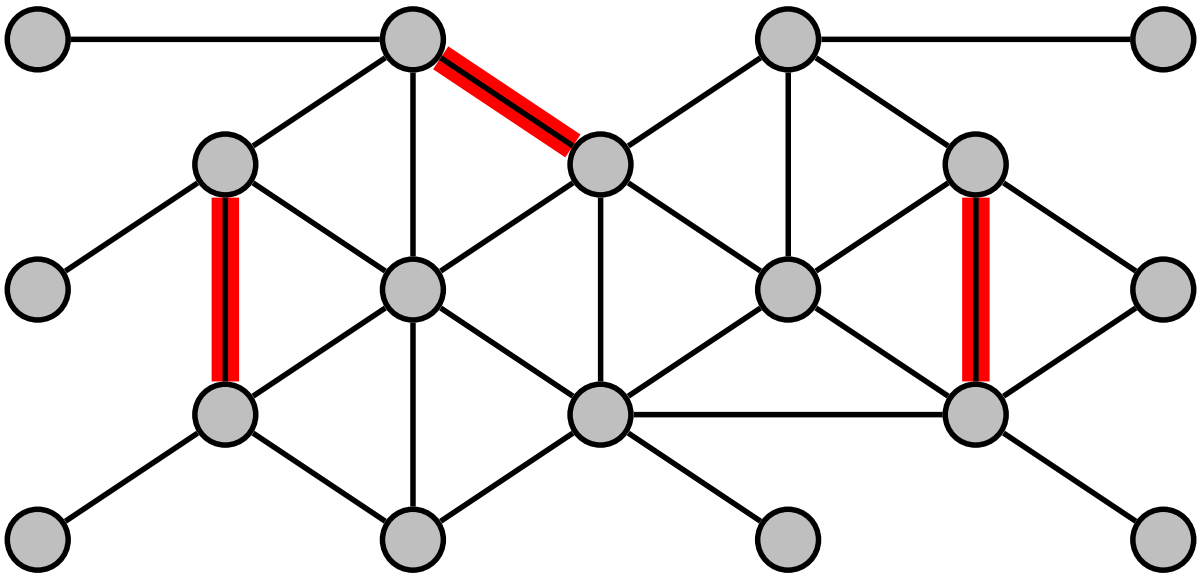
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



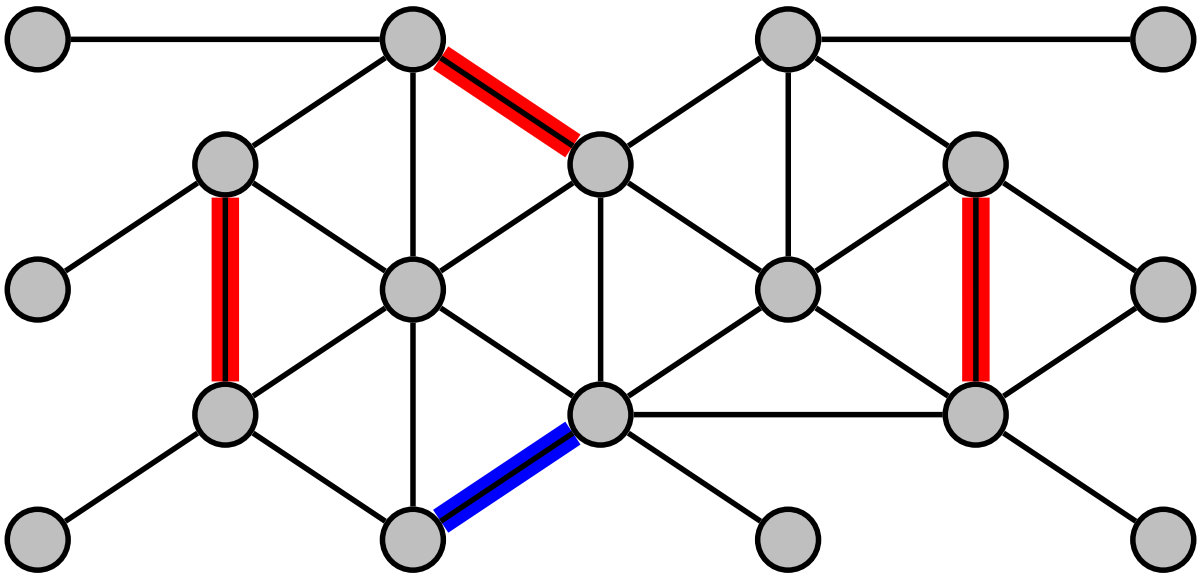
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



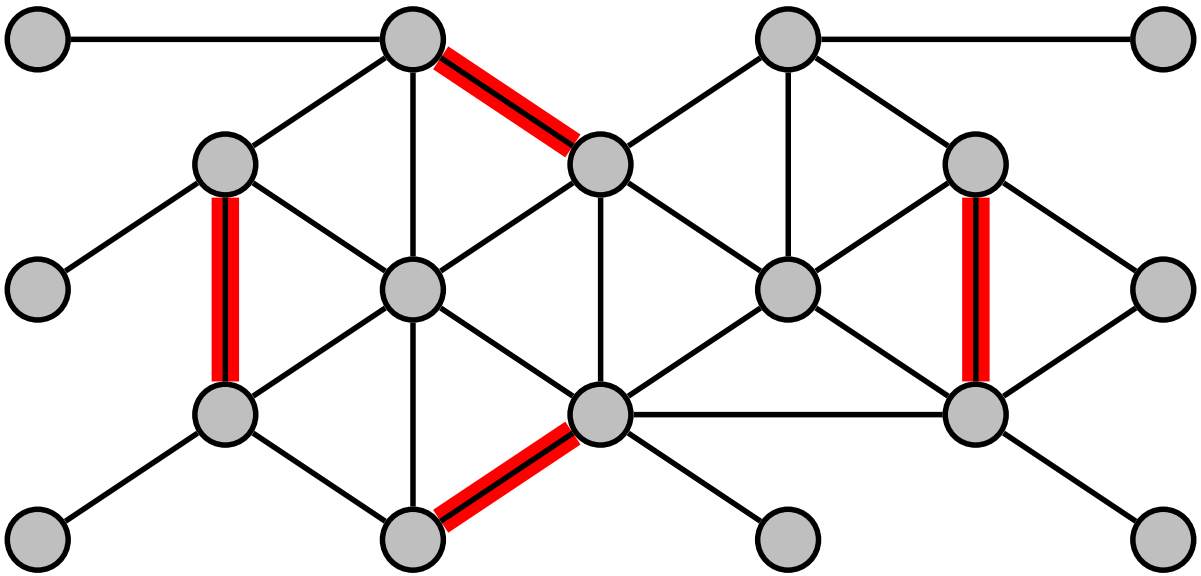
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



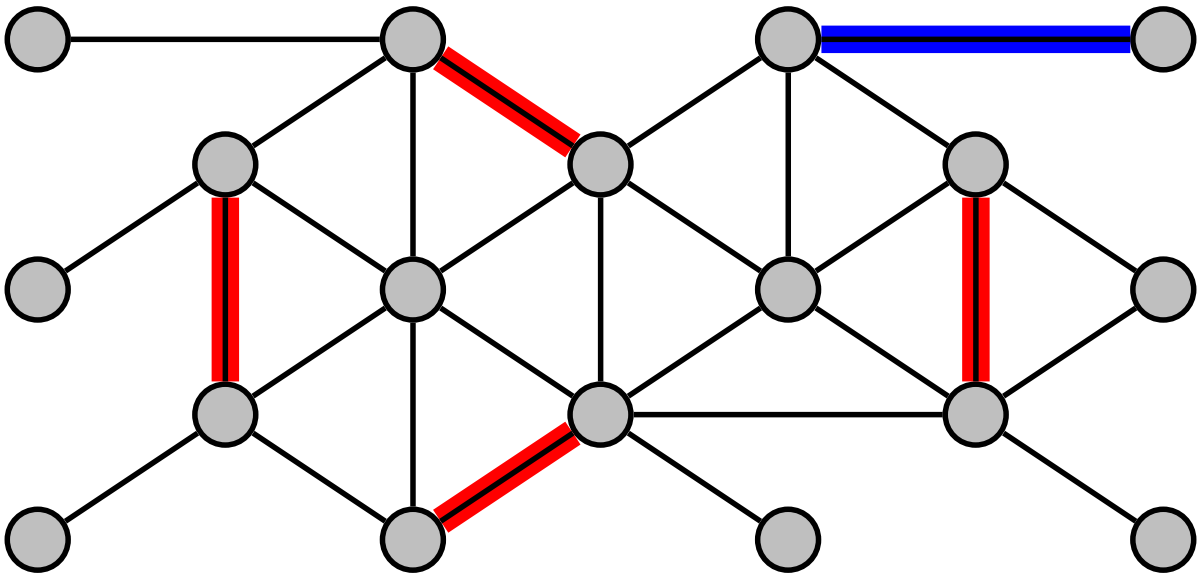
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



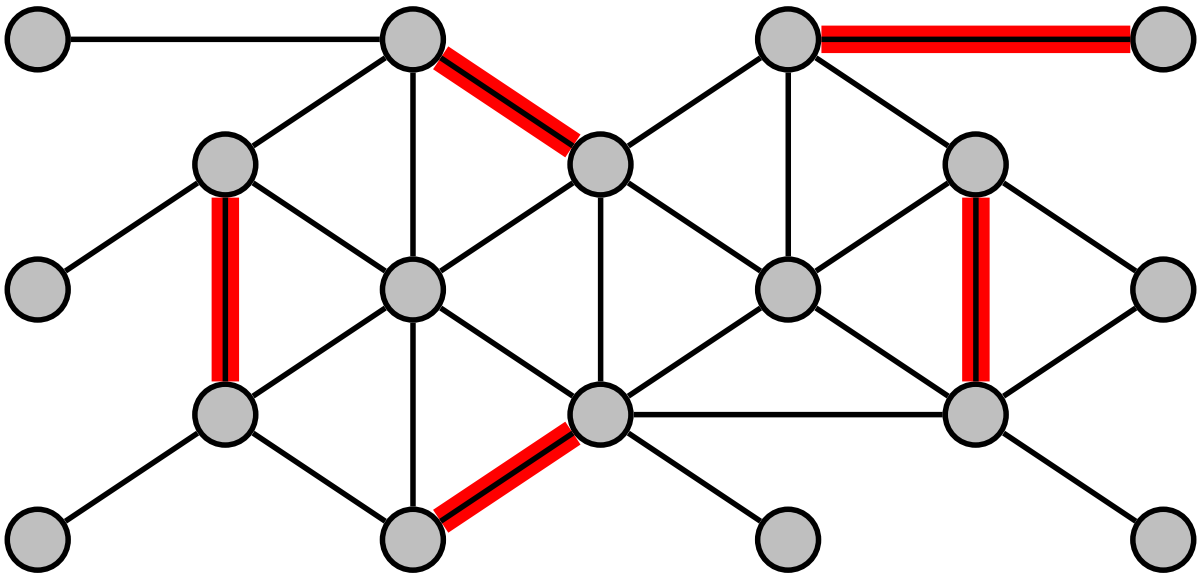
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



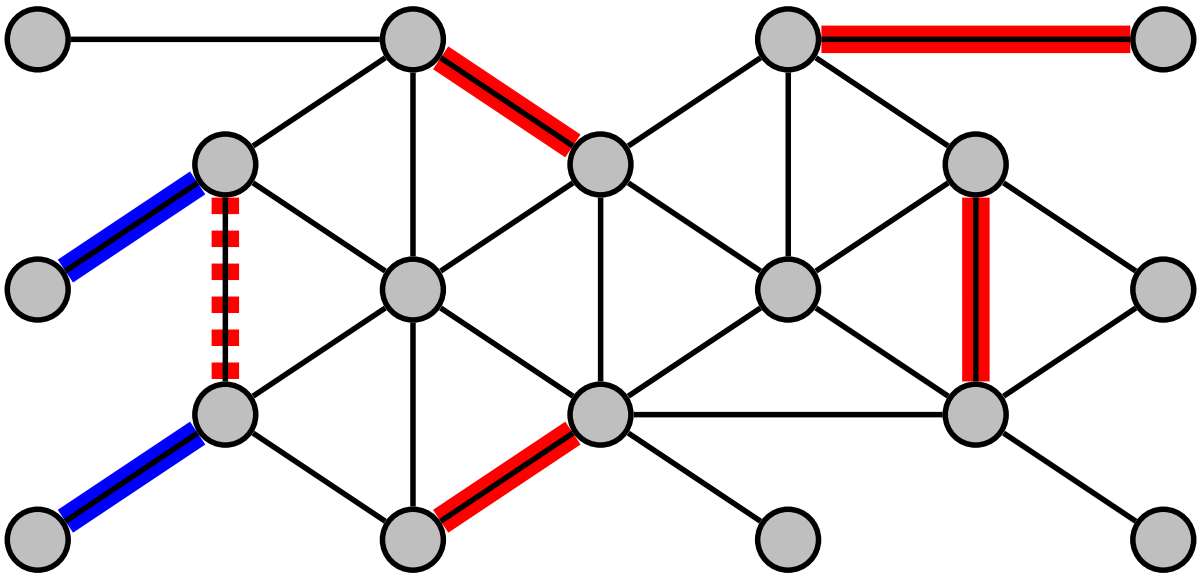
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



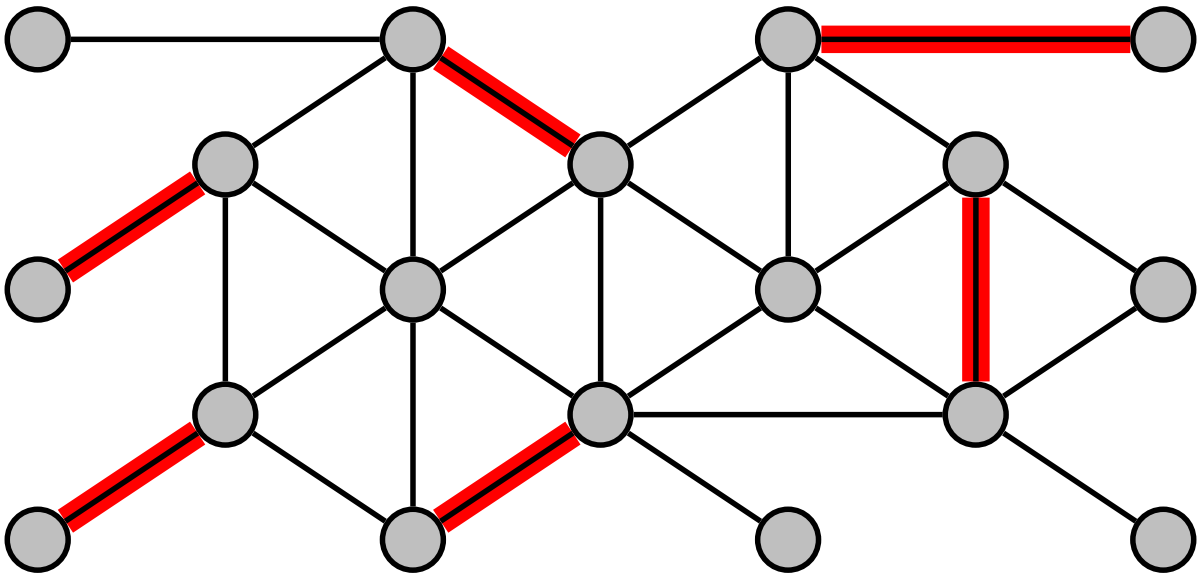
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



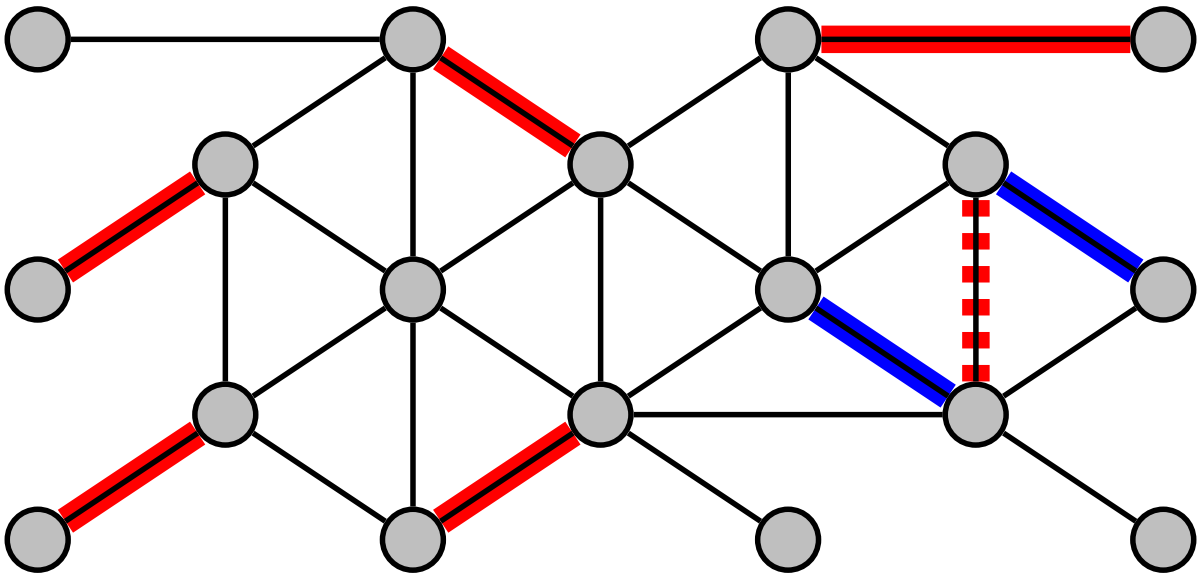
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



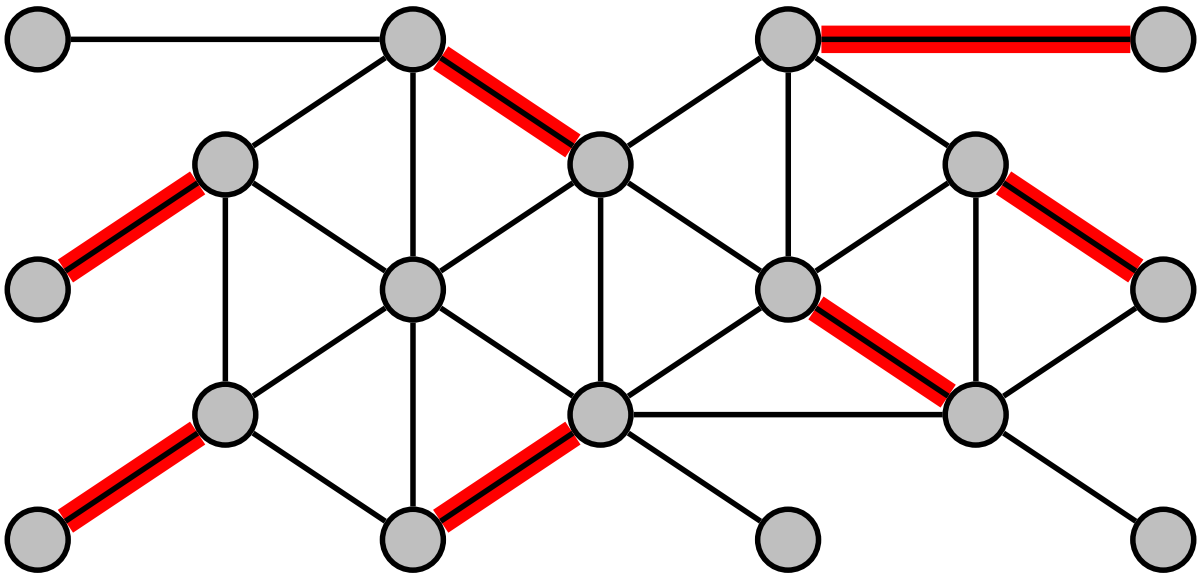
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



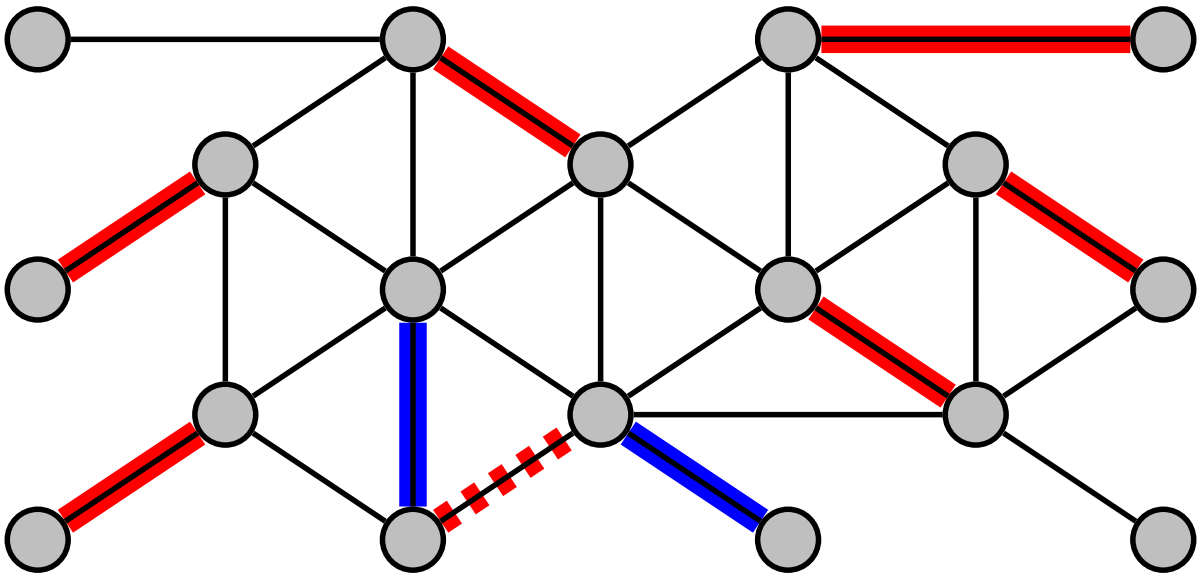
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



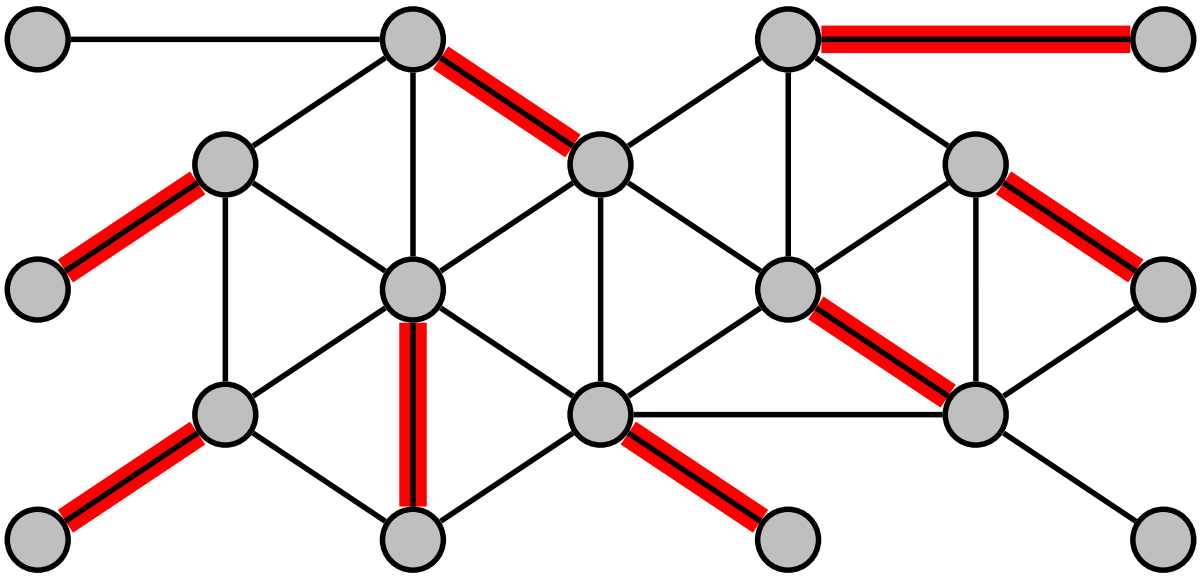
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



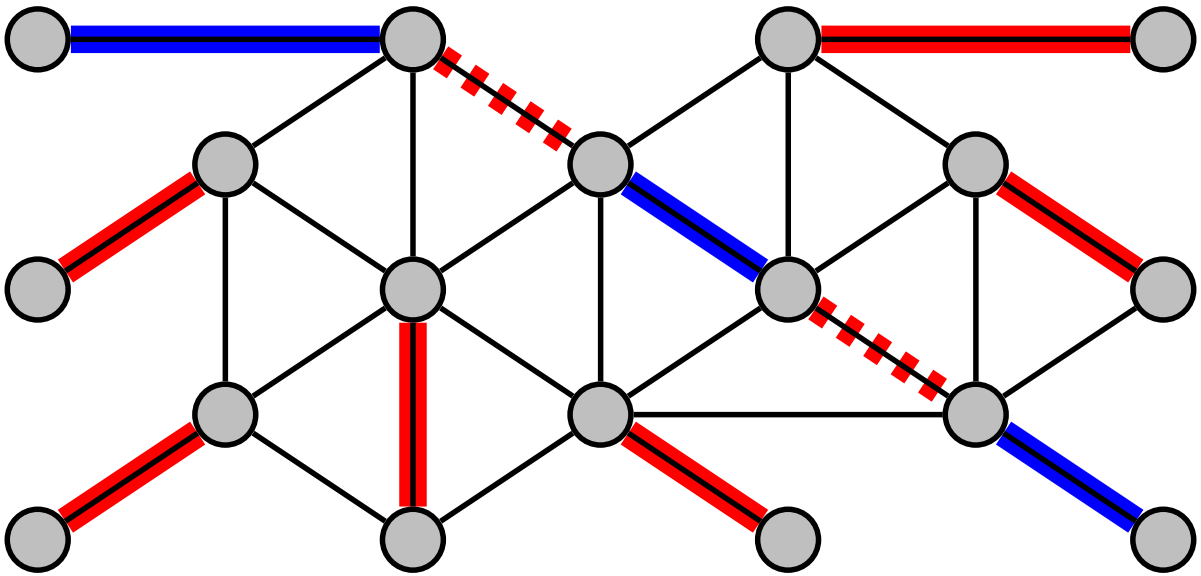
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



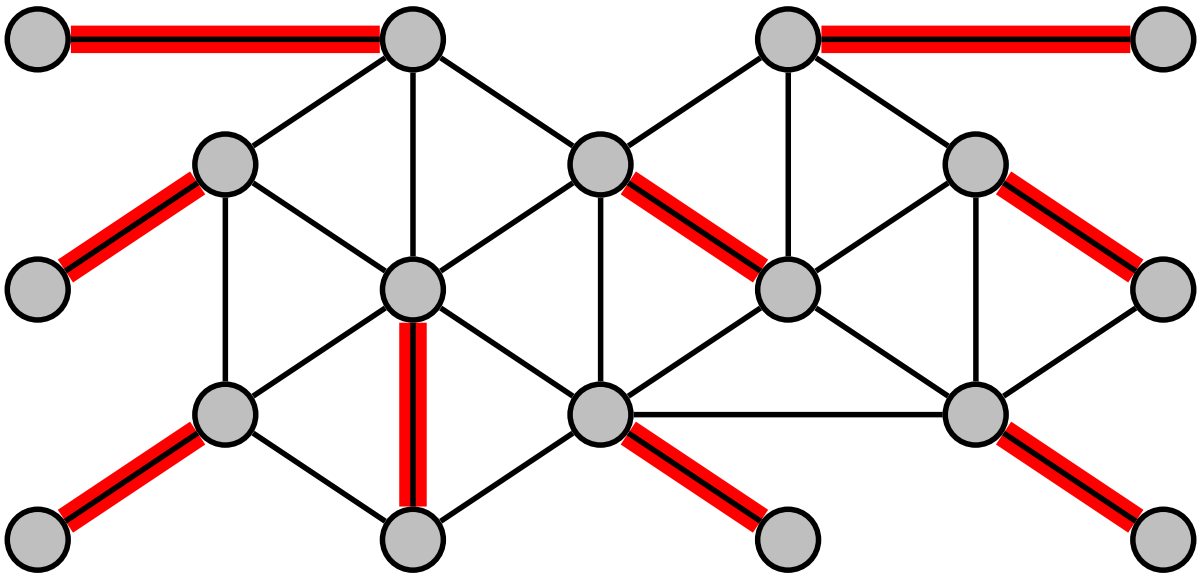
Simulation

Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



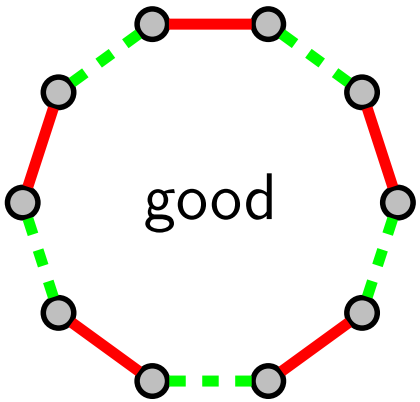
Simulation

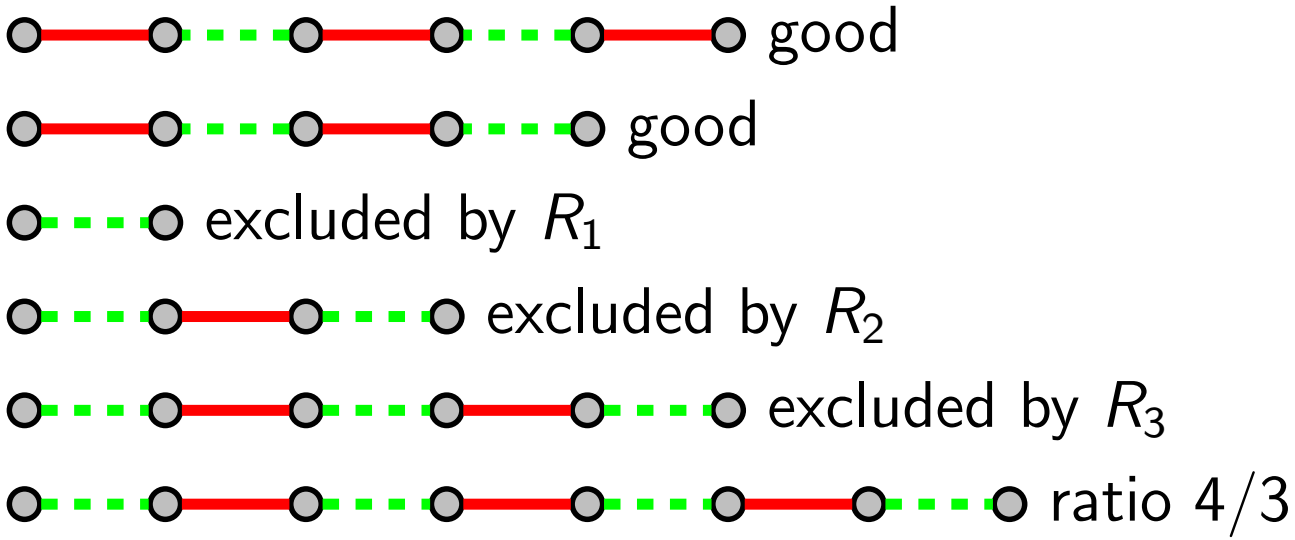
Rule 1: $M = M \cup \{e_1\}$
Rule 2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
Rule 3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$



Analysis

Union of ALG(——) and OPT(- - - -) is a set of paths and even cycles

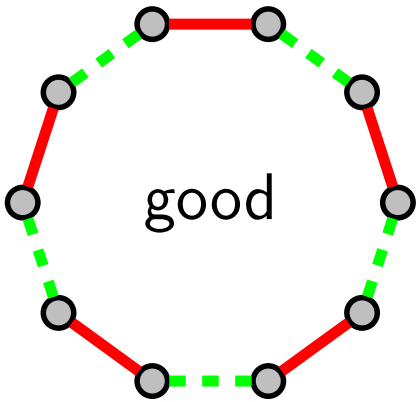


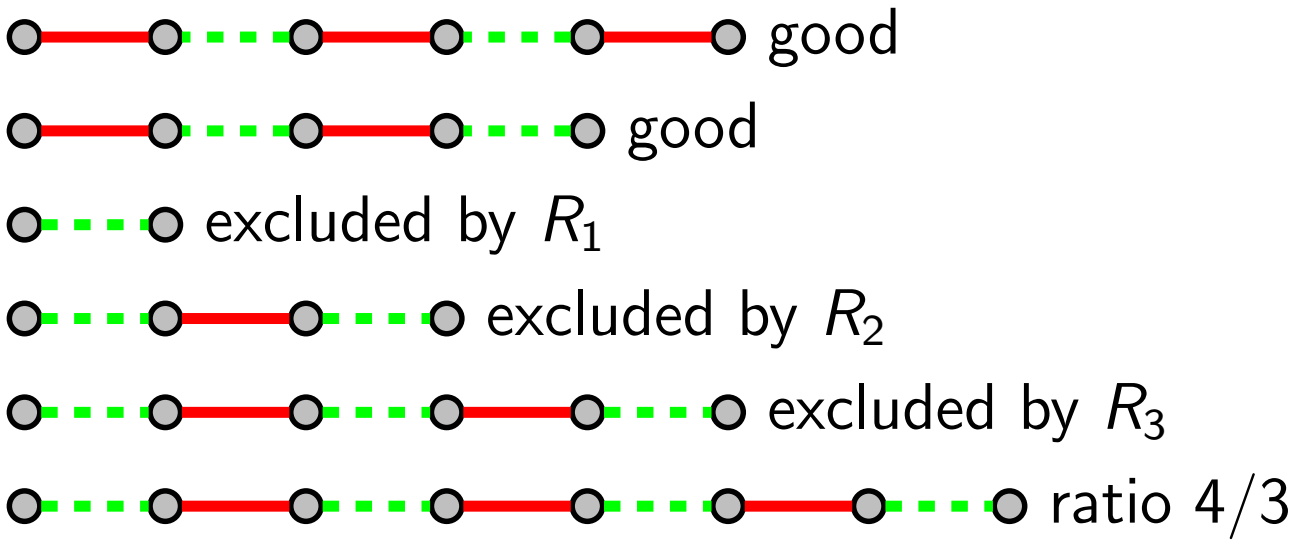


R_1: $M = M \cup \{e_1\}$
R_2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
R_3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$

Analysis

Union of ALG(——) and OPT(- - - -) is a set of paths and even cycles





R_1: $M = M \cup \{e_1\}$
R_2: $M = (M \setminus \{e_1\}) \cup \{e_2, e_3\}$
R_3: $M = (M \setminus \{e_1, e_2\}) \cup \{e_3, e_4, e_5\}$

- $\{R_1\}$ gives 2-apx
- $\{R_1, R_2\}$ gives 3/2-apx
- $\{R_1, R_2, R_3\}$ gives 4/3-apx

Terminology

- Local search: small changes in each step.
- Hill climbing: local search with improving moves only.

Observation

For hill climbing: the larger the neighbourhood, the better local optima, but the longer it takes to explore possible moves.

Local search in approximation

Local search (hill climbing) has been used to derive approximation algorithms for a few problems, such as:

- k-Median,
- k-Means,
- Capacitated Facility Location,
- k-Set Packing.

Outline

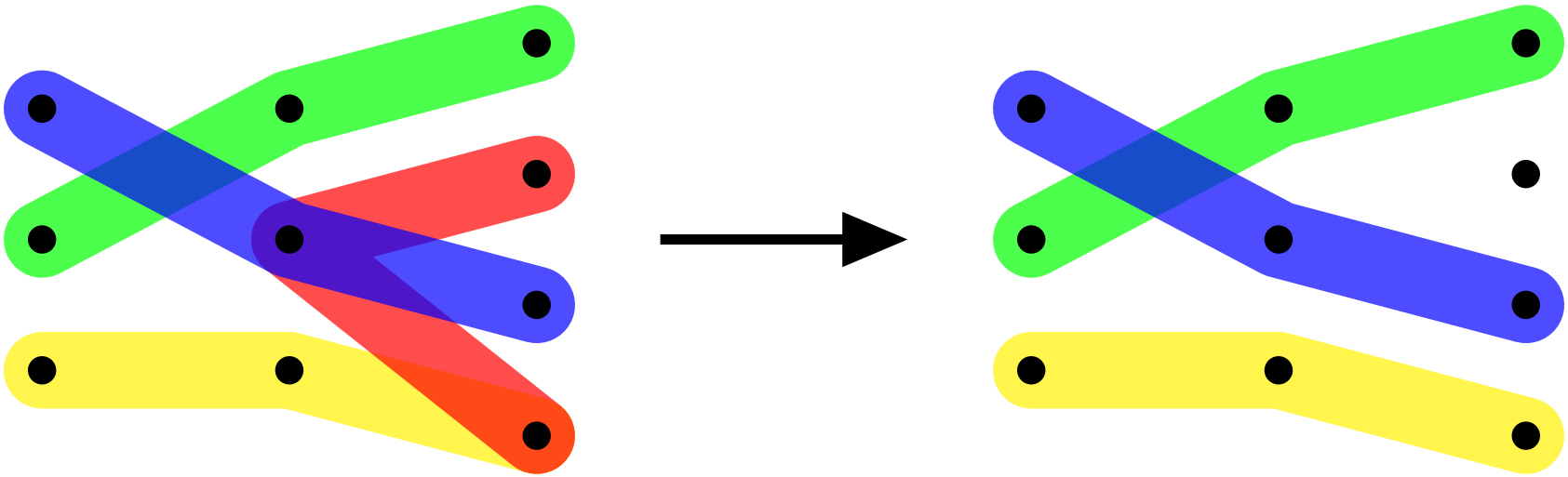
1. k-Set Packing
2. p-Local search
3. Local optima analysis
4. Making local search faster
5. (Strenghtened) LP
6. Weighted k-Set Packing

k-Set Packing

k-Set Packing (k-SP)

Input: a family $\mathcal{F} \subseteq 2^U$ of sets of size at most k.
Goal: find a maximum size subfamily of $\mathcal{F}$ of pairwise disjoint sets.

By n we denote the number of elements of U.



Hardness

- k-Set Packing is NP-hard (among 21 problems of Karp).
- 3-SP is $98/97 - \epsilon$ hard to approximate [Berman and Karpinski 2003].
- For k-SP there is no $O(k/\log k)$-approximation [Hazan et al. 2003].

α-approximation, $\alpha \geqslant 1$

α-approximation algorithm finds $\mathcal{A} \subseteq \mathcal{F}$ of size at least $|OPT|/\alpha$.

Local search for k-SP

p-local search for k-Set Packing

Set $\mathcal{A} = \emptyset$.

While there exists $X \subseteq \mathcal{F} \setminus \mathcal{A}$ such that:

- $|X| \leqslant p$,
- sets in X are disjoint,
- $|X| > |Y|$, where $Y = \{S \in \mathcal{A} : S \cap (\bigcup X) \neq \emptyset\}$.

do $\mathcal{A} := (\mathcal{A} \setminus Y) \cup X$.

We call such X an *improving set*.

Local search for k-SP

- 1-local search:
  $\mathcal{A} := \mathcal{A} \cup \{S\}$
- 2-local search:
  $\mathcal{A} := \mathcal{A} \cup \{S\}$
  $\mathcal{A} := (\mathcal{A} \setminus \{S_1\}) \cup \{S_2, S_3\}$
- 3-local search:
  $\mathcal{A} := \mathcal{A} \cup \{S\}$
  $\mathcal{A} := (\mathcal{A} \setminus \{S_1\}) \cup \{S_2, S_3\}$
  $\mathcal{A} := (\mathcal{A} \setminus \{S_1, S_2\}) \cup \{S_3, S_4, S_5\}$

History of p-local search analysis

Upper bounds for approximation ratio of p-local search for k-SP

author	p	apx ratio
folklore	1	k
folklore	2	$(k+1)/2$
Hurkens and Schrijver'89	$\mathcal{O}(1)$	$(k+\epsilon)/2$
Halldórsson'95	$\mathcal{O}(\log n)$	$(k+2)/3$
C., Grandoni, Mastrolili'13	$\mathcal{O}(\log n)$	$(k+1+\epsilon)/3$

Consequences

$(k+\epsilon)/2$-apx in poly time, $(k+1+\epsilon)/3$-apx in quasipoly time.

History of p-local search analysis

Upper bounds for approximation ratio of p-local search for k-SP

author	p	apx ratio
folklore	1	k
folklore	2	$(k+1)/2$
Hurkens and Schrijver'89	$\mathcal{O}(1)$	$(k+\epsilon)/2$
Halldórsson'95	$\mathcal{O}(\log n)$	$(k+2)/3$
C., Grandoni, Mastrolili'13	$\mathcal{O}(\log n)$	$(k+1+\epsilon)/3$

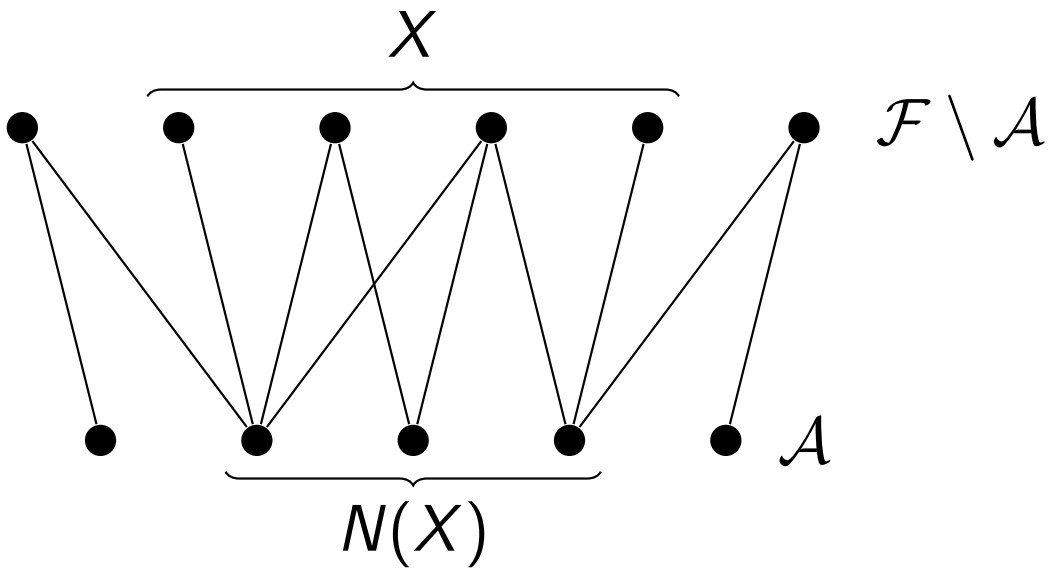
Lower bounds:

Sviridenko and Ward'13	$\Omega(n)$	$k/3$
Fürer and Yu'14	$\Omega(n^{1/5})$	$(k+1)/3$

Local optima analysis

Conflict graph

Recall that $\mathcal{A} \subseteq \mathcal{F}$ is the current solution.
We define a bipartite graph called a *conflict graph*.



We want $X \subseteq \mathcal{F} \setminus \mathcal{A}$ such sets in X are disjoint and $|N(X)| < |X|$.

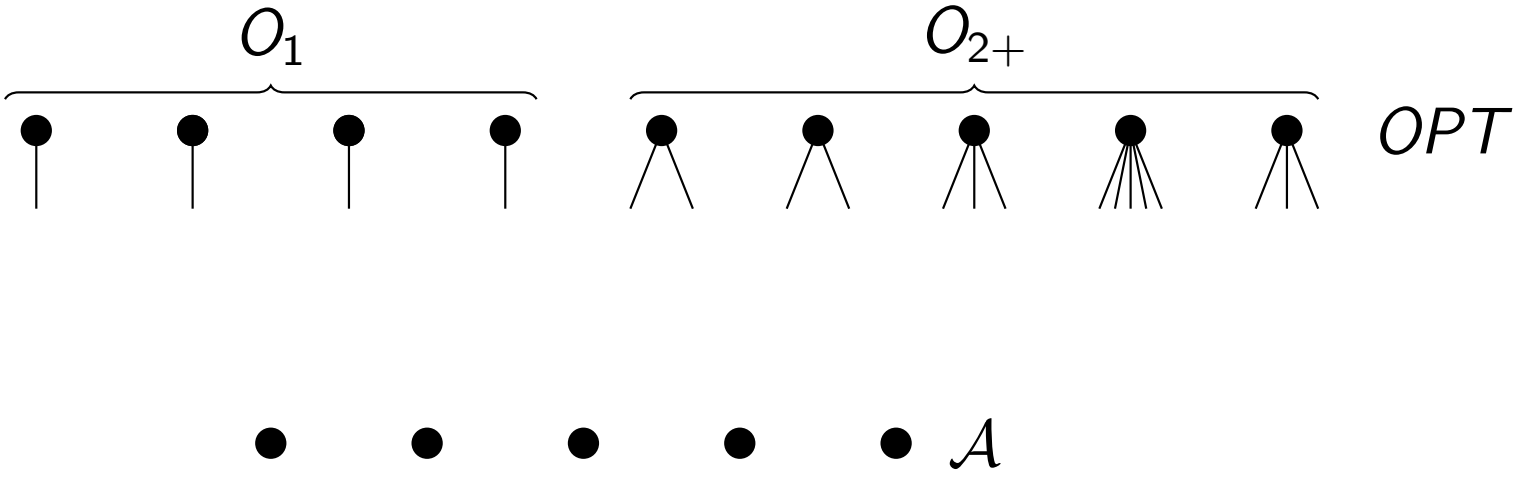
2-Opt Analysis

author	p	apx ratio
folklore	1	k
folklore	2	$(k+1)/2$
Hurkens and Schrijver'89	$\mathcal{O}(1)$	$(k+\epsilon)/2$
Halldórsson'95	$\mathcal{O}(\log n)$	$(k+2)/3$
C., Grandoni, Mastrolili'13	$\mathcal{O}(\log n)$	$(k+1+\epsilon)/3$

$\mathcal{A} := \mathcal{A} \cup \{S\}$
$\mathcal{A} := (\mathcal{A} \setminus \{S_1\}) \cup \{S_2, S_3\}$

2-Opt Analysis

- Let $\mathcal{A}$ be a 2-local search maximum.
- W.l.o.g. assume that $OPT \cap \mathcal{A} = \emptyset$.
- Partition $OPT = O_1 \uplus O_{2+}$ w.r.t. degrees in the subgraph of the conflict graph induced by $\mathcal{A} \cup OPT$.

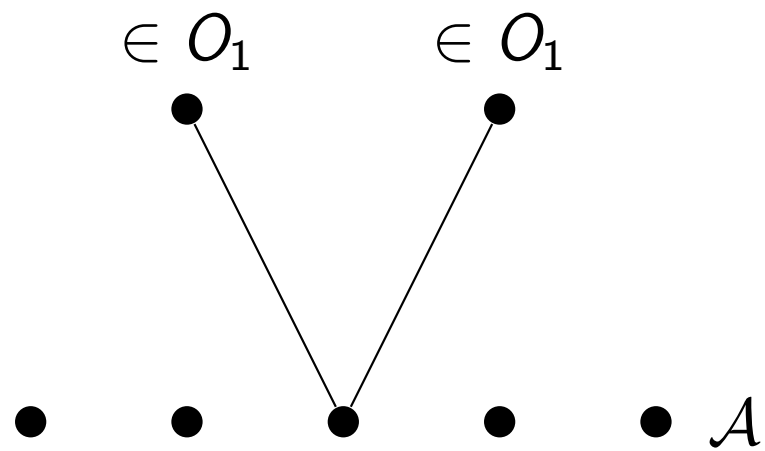


$$|O_1| + 2|O_{2+}| \leqslant |E| \leqslant k|A|$$

2-Opt Analysis

Observation

Two vertices of O_1 cannot be adjacent to the same vertex of $\mathcal{A}$, as they would form an improving set, hence $|O_1| \leqslant |\mathcal{A}|$.



2-Opt Analysis

Summing

$$|O_1| + 2|O_{2+}| \leqslant k|A|$$
$$|O_1| \leqslant |A|$$

we get

$$2|OPT| = 2|O_1| + 2|O_{2+}| \leqslant (k+1)|A|$$

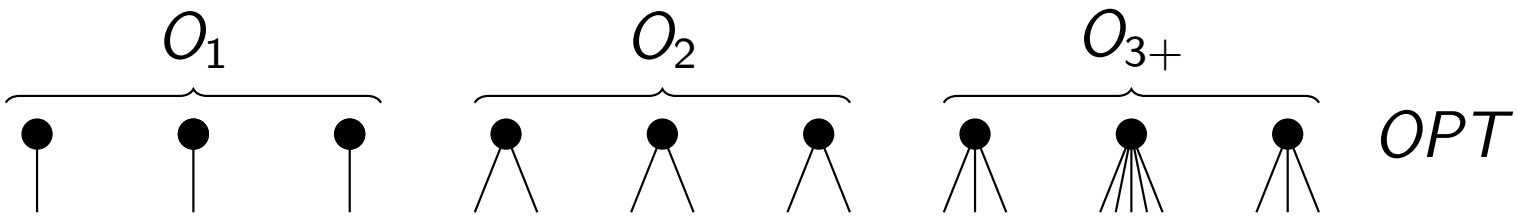
and consequently 2-opt is a $(k+1)/2$-approximation.

Analysis of Halldórsson

author	p	apx ratio
folklore	1	k
folklore	2	$(k+1)/2$
Hurkens and Schrijver'89	$\mathcal{O}(1)$	$(k+\epsilon)/2$
Halldórsson'95	$\mathcal{O}(\log n)$	$(k+2)/3$
C., Grandoni, Mastrolili'13	$\mathcal{O}(\log n)$	$(k+1+\epsilon)/3$

Analysis of Halldórsson

- Define $OPT = O_1 \uplus O_2 \uplus O_{3+}$ w.r.t. degrees in the subgraph of the conflict graph induced by $\mathcal{A} \cup OPT$.



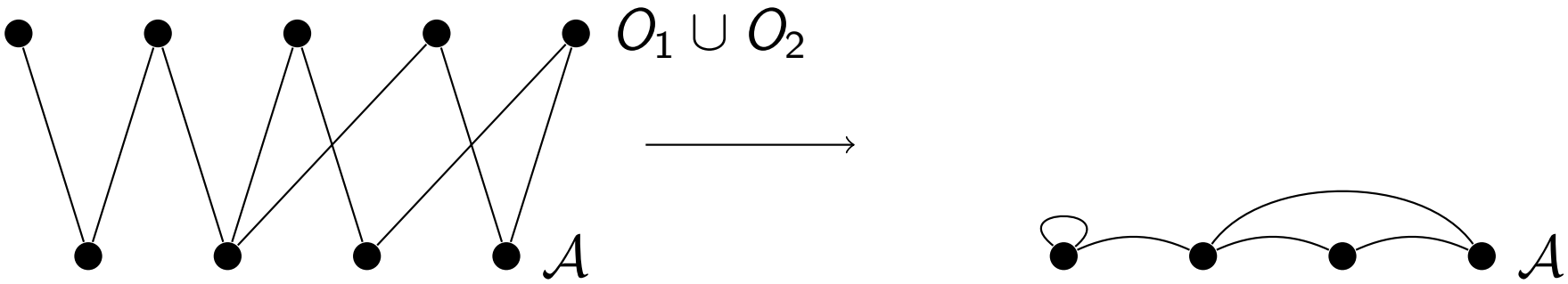
$$|O_1| + 2|O_2| + 3|O_{3+}| \leqslant |E| \leqslant k|A|$$

Analysis of Halldórsson

Halldórsson'95

If $|O_1| + |O_2| > (1+\epsilon)|A|$, there is an improving set of size $c \log n$.

Proof idea:



Lemma

In a graph with $|E|/|V| \geqslant 1+\epsilon$ there is a cycle of length $\leqslant c_\epsilon \log |V|$.

Summing inequalities

Count the number of edges of the conflict graph from both sides.

$$|O_1| + 2|O_2| + 3|O_{3+}| \leqslant |E| \leqslant k|A|$$

Summing up with

$$|O_1| + |O_2| \leqslant (1+\epsilon)|A|$$

$$|O_1| + |O_2| \leqslant (1+\epsilon)|A|$$

gives

$$OPT \leqslant \frac{(k+2+\epsilon)|A|}{3}$$

Making local search faster

First try

- It was known that in quasipolynomial time one can get better approximation ratios.
- Can we run $\mathcal{O}(\log n)$-local search in poly time?
- It would be ideal to have an algorithm finding an improving set of size p in $c^p \mathrm{poly}(n)$ (if it exists).

C.'13

Finding an improving set of size at most p is $W[1]$-hard, i.e. no $f(p)\mathrm{poly}(n)$ time algorithm.

Second try

As it is impossible to look for **all** possible improving sets of size $\mathcal{O}(\log n)$ in poly time, perhaps we can use a different strategy:

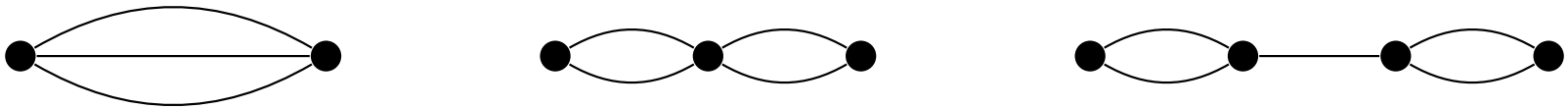
Strategy

1. Inspect what kind of improving sets are sufficient for an approximation algorithm.
2. Show that we can find those efficiently.

What is the structure of the used improving sets?

Halldórsson'95

If $|O_1| + |O_2| > (1+\epsilon)|A|$, then there exists an improving set $X \subseteq \mathcal{F} \setminus \mathcal{A}$ of $\mathcal{O}(\log n)$ size such that $G[N[X]]$ is a subdivision of one of the following graphs.



Sviridenko, Ward'13

Polynomial time $(k+2)/3$-approximation for k-Set Packing.

C.'13

Polynomial time $(k+1+\epsilon)/3$-approximation for k-Set Packing.

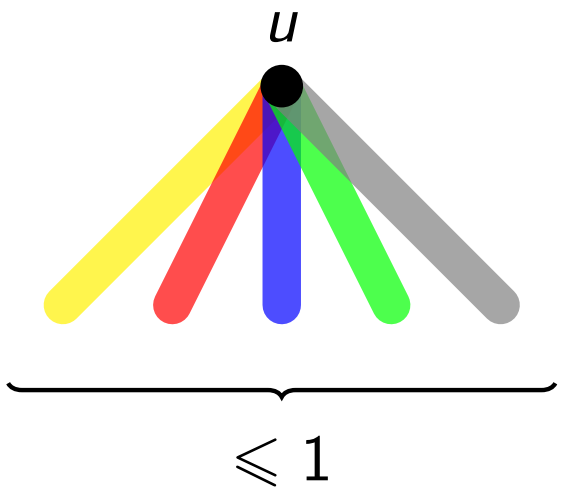
(Strenghtened) Linear Programming

LP

maximize: $\sum_{S \in \mathcal{F}} x_S$

subject to: $\sum_{S \ni u} x_S \leqslant 1 \quad \forall_{u \in U}$

$x_S \geqslant 0 \quad \forall_{S \in \mathcal{F}}$



LP

Füredi 81

Integrality gap of the LP is $k - 1 + 1/k$.

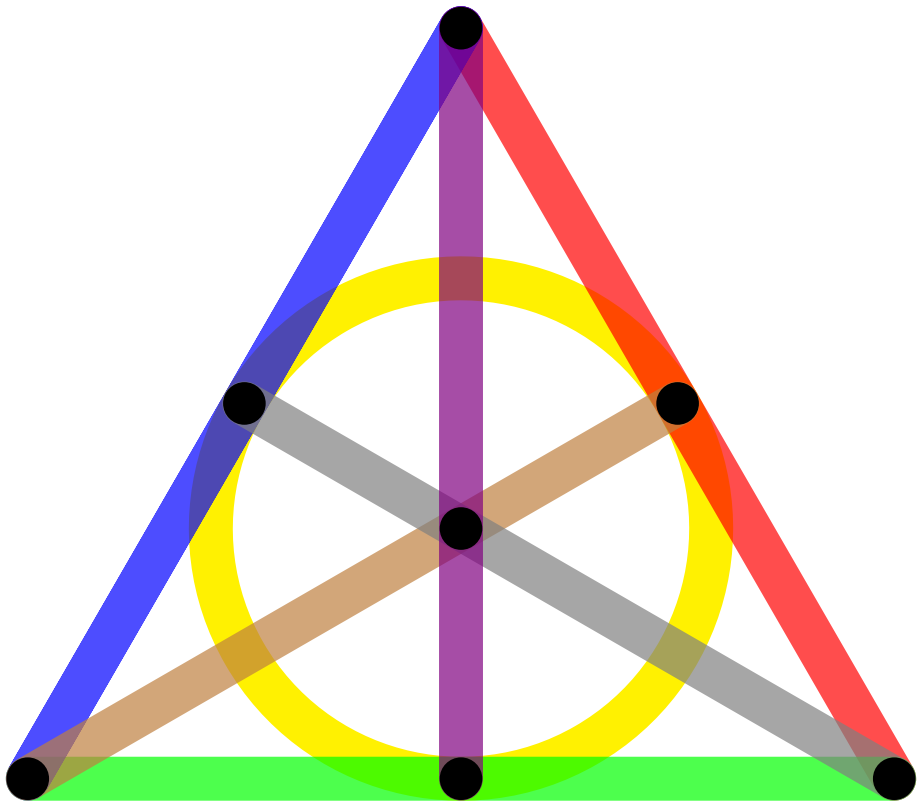
Chan and Lau'10

1. Sherali-Adams level n/k^3 has gap $\geqslant k - 2$.
2. LP + clique constraints has gap $\leqslant (k + 1)/2$.

Consequently Lasserre level 1 has gap $\leqslant (k + 1)/2$.

LP

Integrality gap example for $k = 3$ (Fano plane).



$\forall_S \; x_S = 1/3$

$\sum_S \; x_S = 7/3$

OPT=1

gap=7/3=k-1+1/k

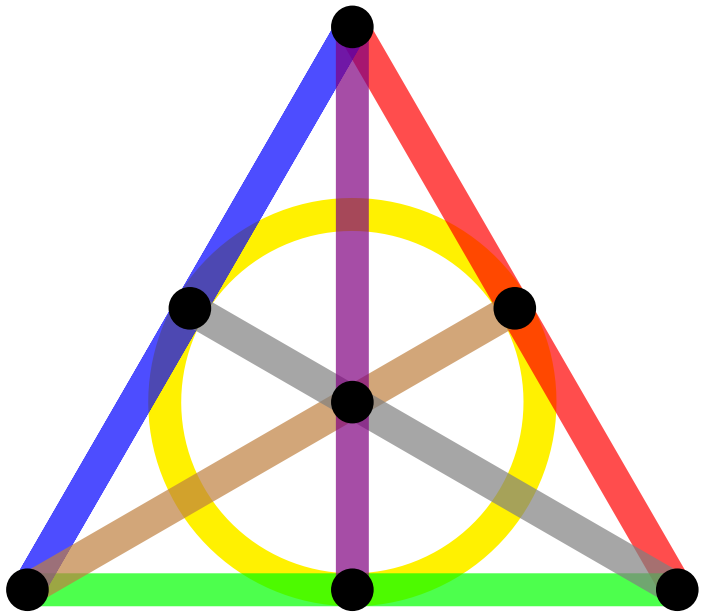
Strenghtened LP

maximize: $\sum_{S \in \mathcal{F}} x_S$

subject to: $\sum_{S \ni u} x_S \leqslant 1 \qquad \forall_{u \in U}$

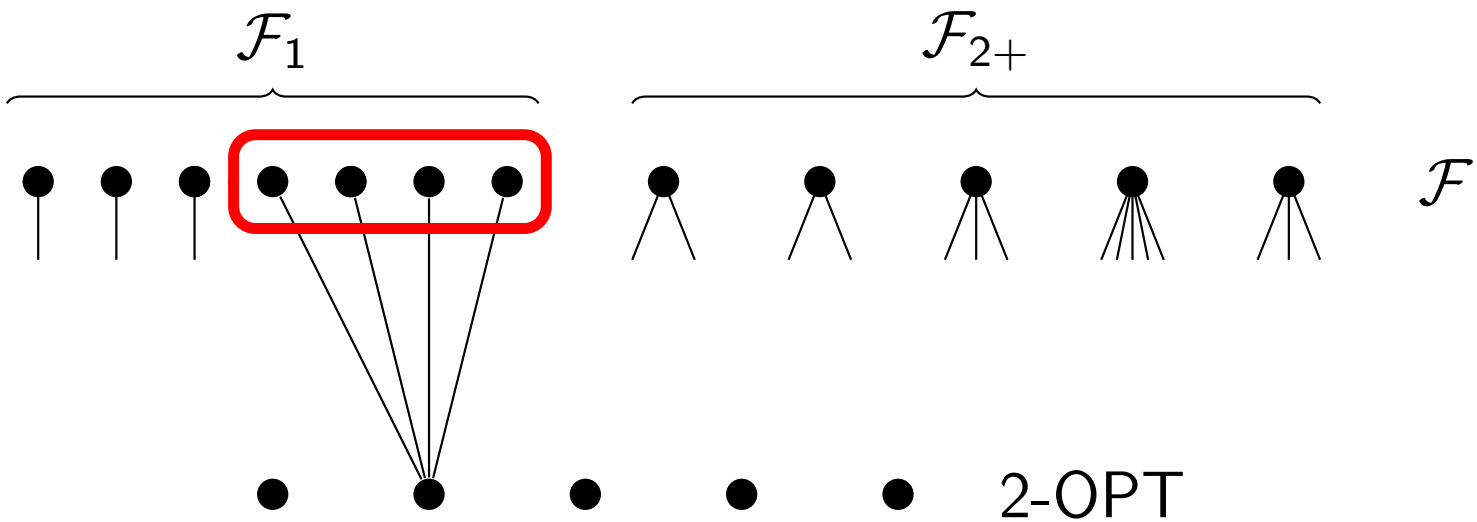
$\sum_{S \in \mathcal{C}} x_S \leqslant 1 \qquad \forall_{\text{clique } \mathcal{C}}$

$x_S \geqslant 0 \qquad \forall_{S \in \mathcal{F}}$



Three different ways of dealing with clique constraints: compact representation, Lovász θ-function, Lasserre level 1.

Strenghtened LP

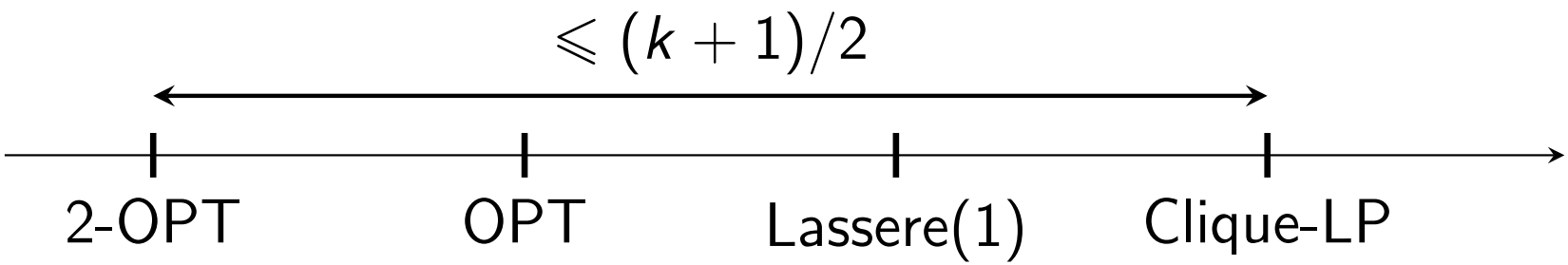


$$x(\mathcal{F}_1) \leqslant |OPT|$$

$$x(\mathcal{F}_1) + x(\mathcal{F}_2) \leqslant k|OPT| \quad \Rightarrow \quad \sum_S x_S \leqslant \frac{(k+1)|OPT|}{2}$$

Strenghtened LP

$$\sum_S x_S \leqslant \frac{(k+1)|OPT|}{2}$$



Strenghtened LP

Open problem

Is there a rounding algorithm for gap $(k+1)/2$?

Open problem

Is there strenghtened LP with integrality gap at most $(k+c)/3$?

Weighted k-Set Packing

Weighted Set Packing

Weighted k-Set Packing

Input: family $\mathcal{F} \subseteq 2^U$ of sets of size at most k,
weight function $\omega : \mathcal{F} \to \mathbb{R}_+$.
Goal: find a maximum weight subfamily of $\mathcal{F}$ of pairwise disjoint sets.

Berman'00

$(k+1+\epsilon)/2$-approximation, $n^{\mathcal{O}(k)}$ time.

- Interesting aspect of Berman's algorithm, optimize $\sum_S w^2(S)$.
- Prusak'16: improved running time to $2^{\mathcal{O}(k^2)} n^{\mathcal{O}(1)}$.
- Open problem: is there $(k+c)/3$-apx?

Motivation

Why is understanding local search algorithms important?

Comparing (even experimentally) local search heuristics is non-trivial.

- Parameter tuning.
- Method variants.
- Tailor made data structures.

Without proper understanding the problem solving process becomes merely an intuition guided random walk.

Questions

Thank you for your attention!