Parameterized Algorithms

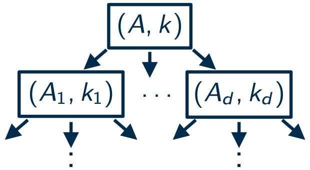
Linear Programs and Kernelization

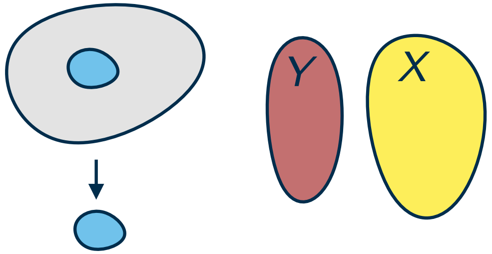
Thomas Bläsius

Content

Basic toolbox

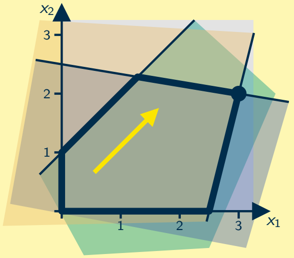
- bounded search trees
- kernelization
- iterative compression

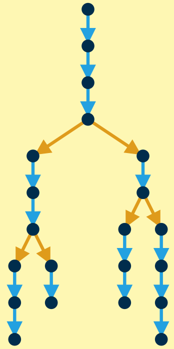


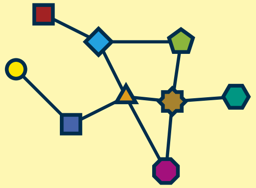


Extended toolbox

- linear programs
- branch-and-reduce
- color coding





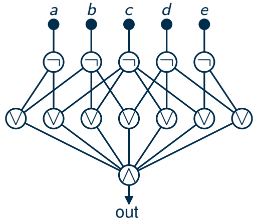


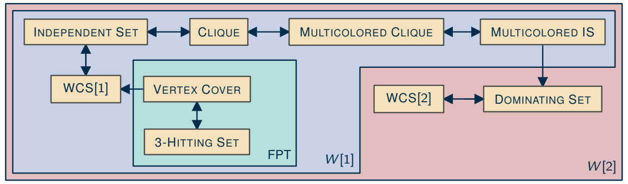
Tree width

- dynamic programming
- chordal and planar graphs
- Courcelle’s theorem

Lower bonds

- kernel lower bounds
- parameterized reductions
- circuits and the W-hierarchy
- ETH and SETH





Example: nutritious and cheap

> *. . . and when Rabbid said, “Honey or condensed milk with your bread?” he was so excited that he said, “Both,” and then, so as not to seem greedy, he added, “But don’t bother about the bread, please.”*
>
> A. A. Milne, Winnie the Pooh

Problem

- burgers do not satisfy the official dietary guidelines
- one serving lacks 0.5 mg vitamin A, 15 mg vit. C, 4 g fiber
- goal: fix this as cheaply as possible
- for this, use carrots, cabbage, and pickles

	carrots	cabbage	pickles
vitamin A (mg/kg)	35	0.5	0.5
vitamin C (mg/kg)	60	300	10
fiber (g/kg)	30	20	10
price (€/kg)	0.75	0.5	0.15

Solution

- let x_1, x_2, and x_3 represents the amount of carrots, cabbage, and pickles, respectively
- optimal solution:
  - 9.5 g carrots
  - 38 g cabbage
  - 290 g pickles

$$\begin{aligned}\text{minimize:}\quad & 0.75x_1 + 0.5x_2 + 0.15x_3\\ \text{side constraints:}\quad & 35x_1 + 0.5x_2 + 0.5x_3 \geq 0.5\\ & 60x_1 + 300x_2 + 10x_3 \geq 15\\ & 30x_1 + 20x_2 + 10x_3 \geq 4\\ & x_i \geq 0\end{aligned}$$

Linear programs

Linear program (LP)

- find real values for variables $x_1, \dots, x_n$
- maximize linear function in the variables
- satisfy side constraints: linear inequalities
- solvable in polynomial time
  (often solved with simplex, which has exponential running time)

Integer linear program (ILP)

- find integer values for $x_1, \dots, x_n$ instead
- NP-hard
- solvers are often fast in practice
- LP-relaxation: ignore restriction to integers

Trivia

- LPs are used since the forties (“program” is a military term for different types of plans)
- first big LP used with the simplex algorithm: optimize cost for a balanced diet of troops
  - 77 variables, 9 constraints
  - simplex algorithm run by humans in 1947: 120 person days
- slightly later using computers: George Dantzig tried to optimize his own diet
  - first try: drink multiple liters of vinegar per day
  - second try: 200 bouillon cubes per day

$\}$ formulating a good LP is harder than solving it

Example: Ice cream for the whole year

Problems when producing ice cream

- the ice consumption d_i heavily depends on the current month i
- changing production causes cost: 50€ per ton change from month i to $i+1$
- storage is expensive: 20€ per ton surplus at the end of each month

Formulating it as an LP

- x_i represents the production in month i
- s_i represents the surplus after month i
- problem: $|x_i - x_{i-1}|$ is not linear
- we need: max of $x_i - x_{i-1}$ and $x_{i-1} - x_i$
- helper variable a_i:
  - constraints ensure that $a_i \geq |x_i - x_{i-1}|$
  - minimizing cost ensures $a_i = |x_i - x_{i-1}|$

enough ice in month i: $x_i + s_{i-1} \geq d_i$

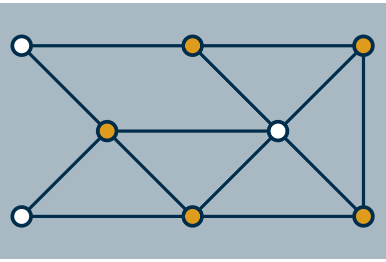
new surplus after month i: $s_i = x_i + s_{i-1} - d_i$

minimize cost: $20 \sum_{i=1}^{12} s_i + 50 \sum_{i=1}^{12} \overset{a_i}{\cancel{|x_i - x_{i-1}|}}$

helper variable: $a_i \geq x_i - x_{i-1}$
$a_i \geq x_{i-1} - x_i$

Best kernelization for VERTEX COVER

Problem: VERTEX COVER
Given a graph $G = (V, E)$ and a parameter k. Does G have a vertex cover of size k?



(vertex set $V' \subseteq V$ with $e \cap V' \neq \emptyset$ for all $e \in E$)

Seen in the first lecture

- chose vertices that “obviously” have to be chosen
- reduce to constant `no`-instance if “obviously” unsolvable

Theorem
VERTEX COVER has a kernel with $O(k^2)$ vertices and edges. It can be computed in $O(m)$ time.

Today

- kernelization via the LP-relaxation of an ILP-formulation of VERTEX COVER
- yields kernel of size $2k$

ILP-Formulation of VERTEX COVER

VERTEX COVER ILP

- variable x_v for $v \in V$ indicates whether v is part of the solution
- minimize size of the vertex cover
- make sure every edge is covered
- make sure that $x_v \in \{0, 1\}$

minimize: $\sum_{v \in V} x_v$

such that: $x_u + x_v \geq 1$ for $uv \in E$

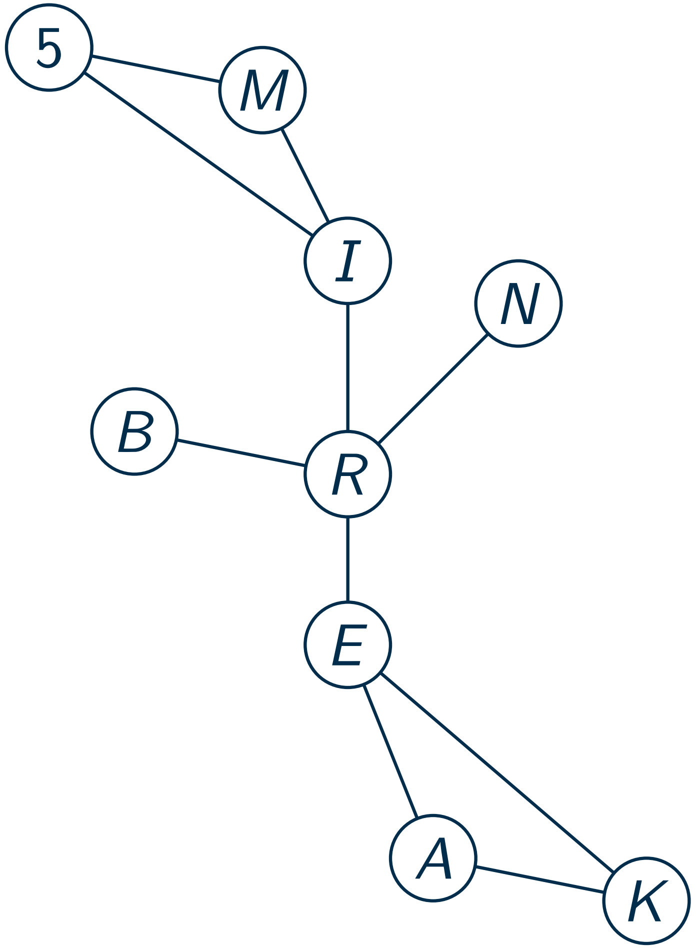
$0 \leq x_v \leq 1$ for $v \in V$

LP-relaxation

- just ignore that we want x_v to be an integer
- vertices might be “selected” partially
- hope: we can still learn something from the LP solution

ILP vs. LP

Find optimal solutions for the ILP and the LP-relaxation



minimize: $\sum_{v \in V} x_v$

such that: $x_u + x_v \geq 1$ for $uv \in E$

$0 \leq x_v \leq 1$ for $v \in V$

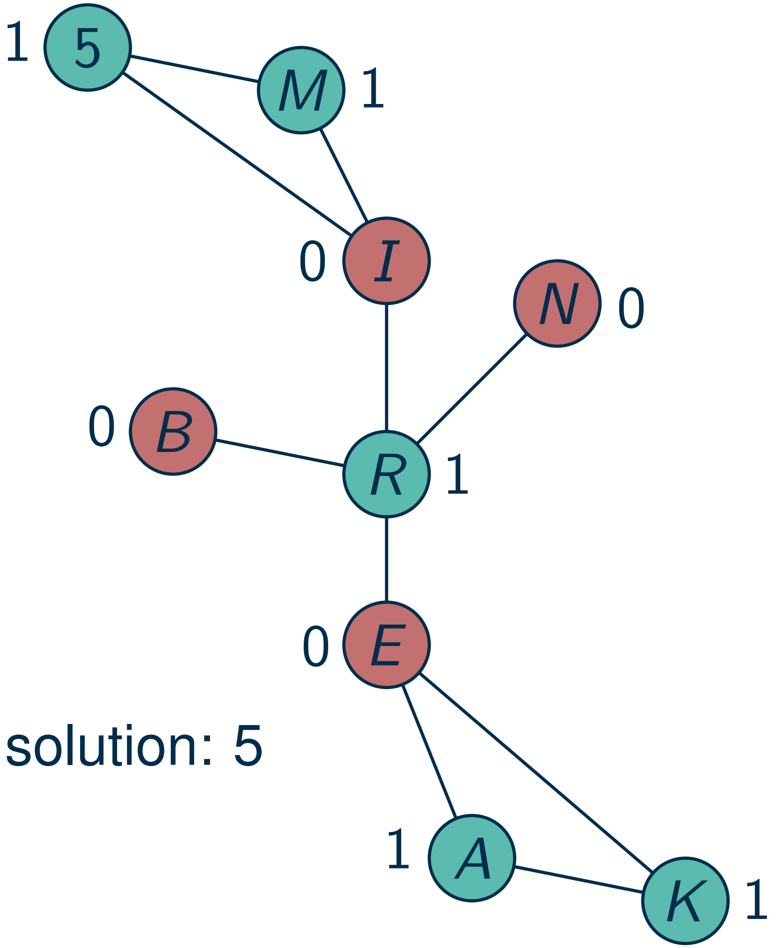
ILP vs. LP

Find optimal solutions for the ILP and the LP-relaxation

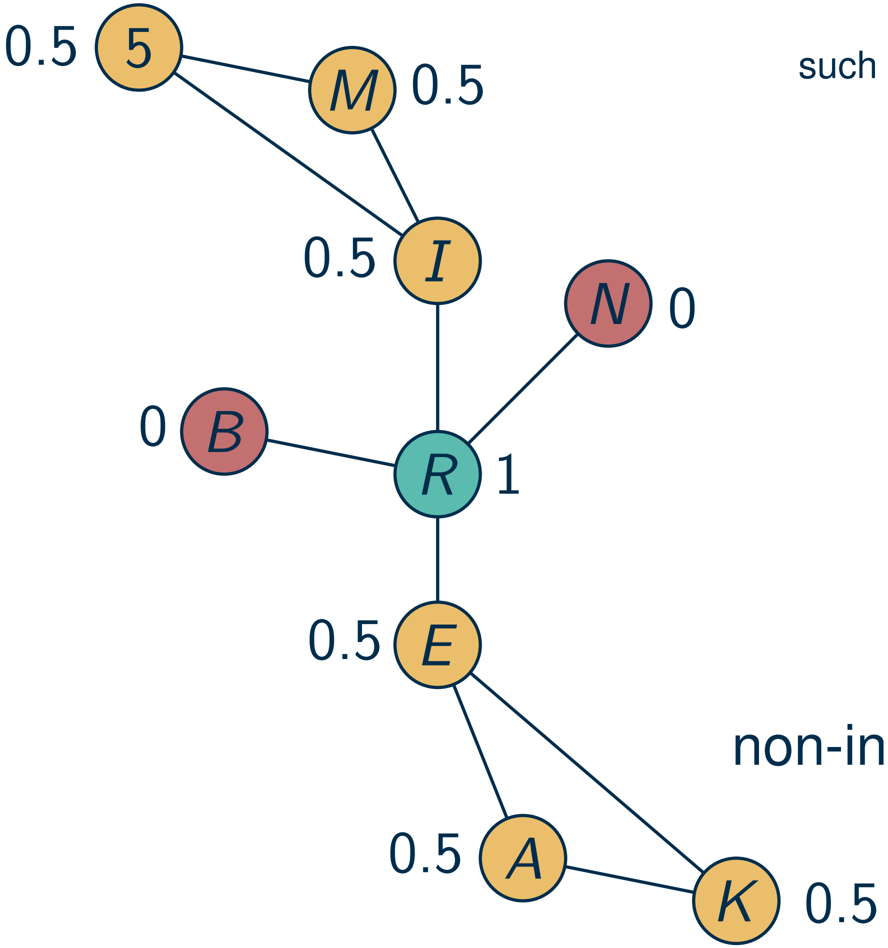
minimize: $\sum_{v \in V} x_v$

such that: $x_u + x_v \geq 1$ for $uv \in E$

$0 \leq x_v \leq 1$ for $v \in V$



integer solution: 5



non-integer solution: 4

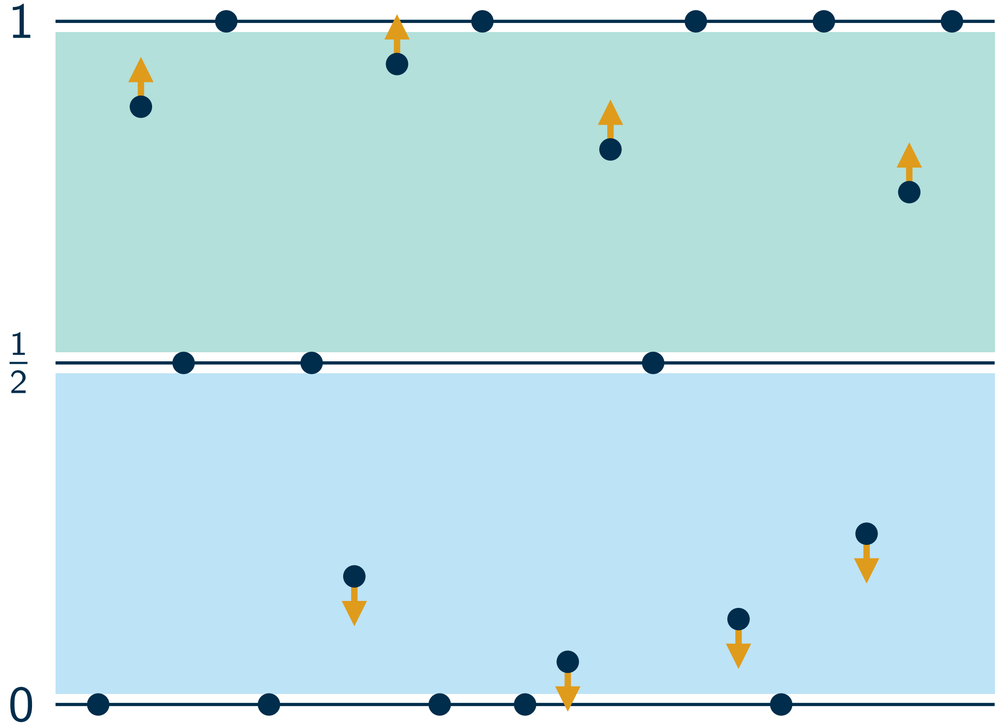
Almost integer solutions

Lemma
The LP-relaxation has an optimal solution with $x_v \in \{0, \frac{1}{2}, 1\}$ for all $v \in V$. Such a solution can be found in polynomial time.

minimize: $\sum_{v \in V} x_v$

such that: $x_u + x_v \geq 1$ for $uv \in E$
$0 \leq x_v \leq 1$ for $v \in V$

Proof
- consider an optimal solution that is not half-integral
- let $A = \{x \mid 0 < x < \frac{1}{2}\}$ and $B = \{x \mid \frac{1}{2} < x < 1\}$ and assume $|A| \geq |B|$ ($|A| \leq |B|$ is symmetric)
- decrease all $x \in A$ and increase all $x \in B$ by ε
  - the total cost does not increase
  - $0 \leq x_v \leq 1$ remains true (for small enough ε)
  - $x_u + x_v \geq 1$ remains true for $uv \in E$
- iteratively decrease number of vars not half-integral



Plan for the following

Lemma
The LP-relaxation has an optimal solution with $x_v \in \{0, \frac{1}{2}, 1\}$ for all $v \in V$. Such a solution can be found in polynomial time.

minimize: $\sum_{v \in V} x_v$

such that: $x_u + x_v \geq 1$ for $uv \in E$

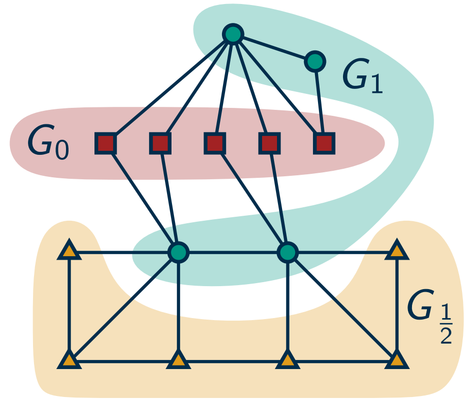
$0 \leq x_v \leq 1$ for $v \in V$

Kernelization

- half-integral solution splits V into subsets V_0, $V_{\frac{1}{2}}$, and V_1 → three subgraphs G_0, $G_{\frac{1}{2}}$, G_1
- show: V_1 plus optimal solution for $G_{\frac{1}{2}}$ is optimal solution for G → $G_{\frac{1}{2}}$ serves as kernel
- kernel size: if $|V_{\frac{1}{2}}| > 2k$ then no solution of size k exists

Why?

Lemma
Let $S_{\frac{1}{2}}$ be a vertex cover of $G_{\frac{1}{2}}$. Then $S_{\frac{1}{2}} \cup V_1$ is a VC of G.
If $S_{\frac{1}{2}}$ is a minimum VC of $G_{\frac{1}{2}}$, then $S_{\frac{1}{2}} \cup V_1$ is a minimum VC of G.



$S_{\frac{1}{2}} \cup V_1$ is a vertex cover

Lemma
Let $S_{\frac{1}{2}}$ be a vertex cover of $G_{\frac{1}{2}}$. Then $S_{\frac{1}{2}} \cup V_1$ is a VC of G.
If $S_{\frac{1}{2}}$ is a minimum VC of $G_{\frac{1}{2}}$, then $S_{\frac{1}{2}} \cup V_1$ is a minimum VC of G.

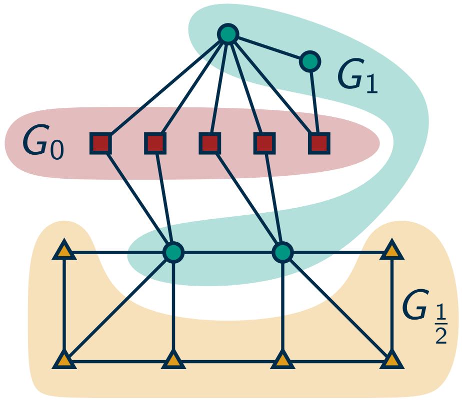
minimize: $\sum_{v \in V} x_v$

such that: $x_u + x_v \geq 1$ for $uv \in E$

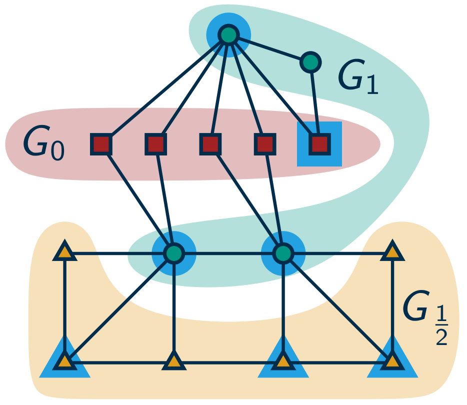
$0 \leq x_v \leq 1$ for $v \in V$

Proof

- consider an edge $uv \in E$
- **Case 1:** $u \in V_1$ oder $v \in V_1 \Rightarrow uv$ is covered
- **Case 2:** $u, v \in V_{\frac{1}{2}} \Rightarrow u \in S_{\frac{1}{2}}$ or $v \in S_{\frac{1}{2}} \Rightarrow uv$ is covered
- there are no further cases, as otherwise $x_u + x_v < 1$



Minimality of $S_{\frac{1}{2}} \cup V_1$



Lemma

Let $S_{\frac{1}{2}}$ be a vertex cover of $G_{\frac{1}{2}}$. Then $S_{\frac{1}{2}} \cup V_1$ is a VC of G.

If $S_{\frac{1}{2}}$ is a minimum VC of $G_{\frac{1}{2}}$, then $S_{\frac{1}{2}} \cup V_1$ is a minimum VC of G.

Proof

- consider any vertex cover $S^\star = S^\star_{\frac{1}{2}} \cup S^\star_0 \cup S^\star_1$ and show $|S_{\frac{1}{2}} \cup V_1| \leq |S^\star|$
- $S^\star_{\frac{1}{2}}$ is VC of $G_{\frac{1}{2}}$ $\Rightarrow |S_{\frac{1}{2}}| \leq |S^\star_{\frac{1}{2}}|$ → it remains to show: $|V_1| \leq |S^\star_0| + |S^\star_1| \Leftrightarrow |V_1 \setminus S^\star_1| \leq |S^\star_0|$

New solution for the LP

$$x'_v = \begin{cases} \frac{1}{2} & \text{if } v \in V_1 \setminus S^\star_1 \cup S^\star_0 \\ x_v & \text{otherwise} \end{cases}$$

Why is this a solution for the LP?

It follows:

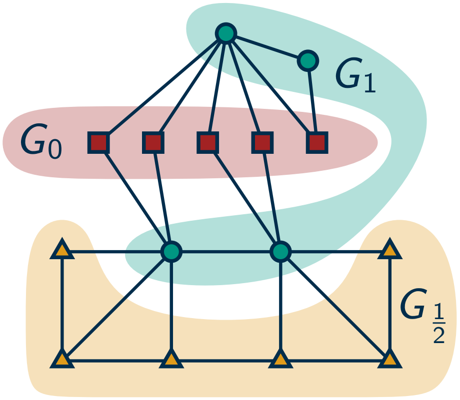
$$\sum_{v \in V} x'_v = \sum_{v \in V} x_v \underbrace{- \frac{|V_1 \setminus S^\star_1|}{2} + \frac{|S^\star_0|}{2}}_{\geq 0 \text{ as } (x_v)_{v \in V} \text{ is optimal}} \Rightarrow |V_1 \setminus S^\star_1| \leq |S^\star_0|$$

Kernelization

minimize: $\sum_{v \in V} x_v$

such that: $x_u + x_v \geq 1$ for $uv \in E$

$0 \leq x_v \leq 1$ for $v \in V$



Reduction rule

- solve the LP relaxation $\rightarrow (x_v)_{v \in V}$
- if $\sum_{v \in V} x_v > k$: constant size `no`-instance
- otherwise: delete $V_0 \cup V_1$ and reduce k by $|V_1|$

Lemma: The reduction rule is safe.

Proof

- optimal solution of the ILP not smaller than optimal solution of the LP relaxation
- $\sum_{v \in V} x_v > k \Rightarrow$ no VC of size k exists
- previous lemma: there is a minimum VC that contains V_1 and excludes V_0

Theorem
For VERTEX COVER, a kernel with at most $2k$ vertices can be computed in $O(??)$ time.

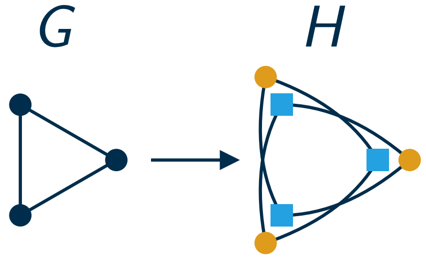
Note: $\frac{1}{2}|V_{\frac{1}{2}}| \leq \sum_{v \in V} x_v \leq k \Rightarrow |V_{\frac{1}{2}}| \leq 2k$

Running time: dominated by solving the LP

Get rid of the LP → better running time

Construction of a helper graph *H*

- split each vertex v into v_o (orange) and v_b (blue)
- edge uv in G → edges $u_o v_b$ and $u_b v_o$ in H



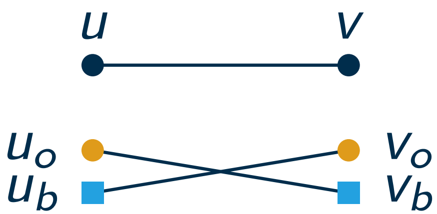
minimize: $\sum_{v \in V} x_v$

such that: $x_u + x_v \geq 1$ for $uv \in E$
$0 \leq x_v \leq 1$ for $v \in V$

Lemma
the LP-relaxation has a solution of value $\sum_{v \in V} x_v = k \Leftrightarrow H$ has a vertex cover S of size $2k$

LP-solution → VC

- $x_v = 1$ → select v_o and v_b
- $x_v = \frac{1}{2}$ → select v_o
- $x_v = 0$ → select nothing



VC → LP-solution

- both v_o and v_b selected → $x_v = 1$
- either v_o or v_b selected → $x_v = \frac{1}{2}$
- both not selected → $x_v = 0$

Computing a VC in *H*: H is bipartite ⇒ a minimum VC can be computed in $O(m\sqrt{n})$ (exercise)

Wrap-Up

(Integer) linear programs

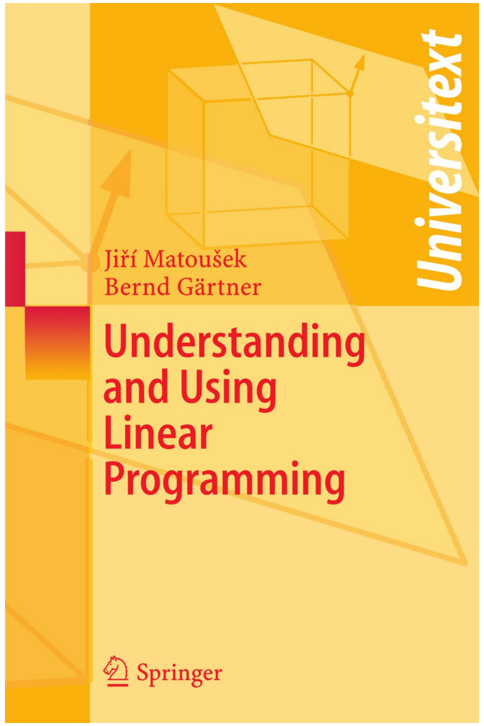
- easy way to model other problems
- sometimes additional variables help
- LP relaxations can be a useful tool
- next lecture: duality

Kernelization via (I)LP

- the ILP for VERTEX COVER is somewhat well-behaved
- use solution of the LP-relaxation for kernelization
- solve the LP-relaxation via VERTEX COVER in bipartite graphs

Theorem
For VERTEX COVER, a kernel with at most $2k$ vertices can be computed in $O(m\sqrt{n})$ time.

Literature



Understanding and Using Linear Programming

- exceptionally well written and compact book
- free for KIT members

`link.springer.com/book/10.1007/978-3-540-30717-4`

Integer Programming in Parameterized Complexity: Three Miniatures

- Tomás Gavenciak, Dusan Knop, Martin Koutecký [2019]
- good overview about ILPs for parameterized problems; has many additional references

`drops.dagstuhl.de/opus/volltexte/2019/10222/`