Updated 4/6/2022

Topic 3: Efficient query evaluation
Unit 2: Cyclic queries (continued)
Lecture 21

Wolfgang Gatterbauer

CS7240 Principles of scalable data management (sp22)

https://northeastern-datalab.github.io/cs7240/sp22/

4/5/2022

Outline: T3-2: Cyclic conjunctive queries

- T3-1: Acyclic conjunctive queries
- T3-2: Cyclic conjunctive queries
  - 2SAT (a detour)
  - Tree decompositions
  - Decompositions of hypertrees
  - Duality in Linear programming (a quick primer)
  - AGM bound (maximal result size for full CQs)
  - Worst-case optimal joins & the triangle query
  - Worst-case optimal joins & the 4-cycle
  - Optimal joins & the 4-cycle

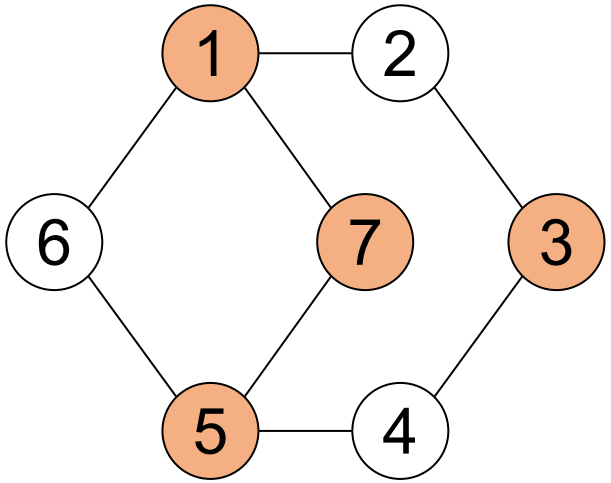
Topic Duality in Linear Programming (LP)

- Subtopics
  - Connections between Set packing and Set covers in graphs
  - Linear Programming (LP) and duality gaps
  - LP relaxations of ILP problems (Integer Linear Programming)
  - Duality b/w independent vertex sets and edge covers

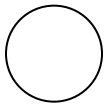
Let's use graphs to explain duality in LP (Linear Programming)

- Coverings problems in graphs: set of subsets that cover all elements
  - min set cover: min number of subsets to cover the entire domain
  - min vertex cover: min number of vertices to cover all edges
  - min edge cover: min number of edges to cover all vertices
- Packing problems in graphs: set of disjoint subsets
  - max set packing: max number of subsets that are pairwise disjoint
  - max independent (vertex) set: max number of vertices not sharing edges
  - max independent edge set = matching: maximum number of edges that don't share any nodes (every vertex can be in max one matching)

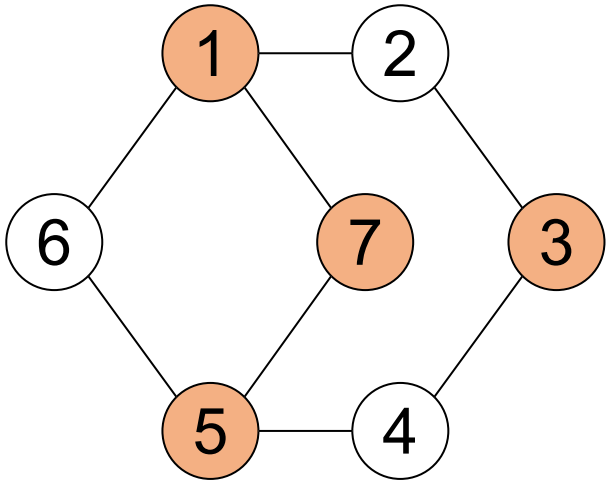
Independent set



Independent set (IS): set of vertices that are not connected (white)

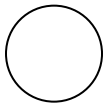


VC vs. Ind set ?



Independent set (IS): set of vertices that are not connected (white)

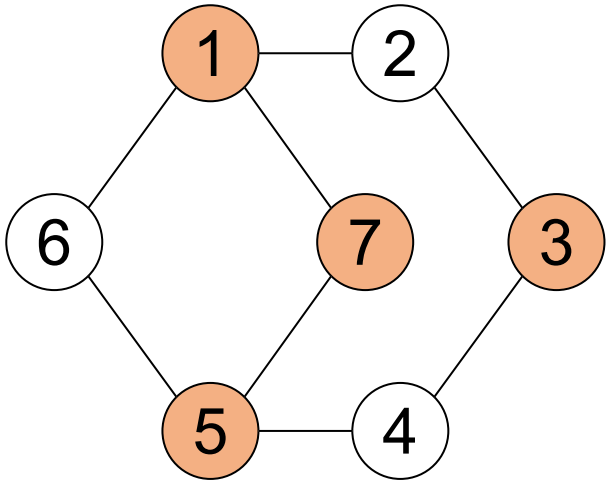
Vertex cover (VC): set of vertices that covers all edges



Assume you are given an independent set.
How do you find a vertex cover?

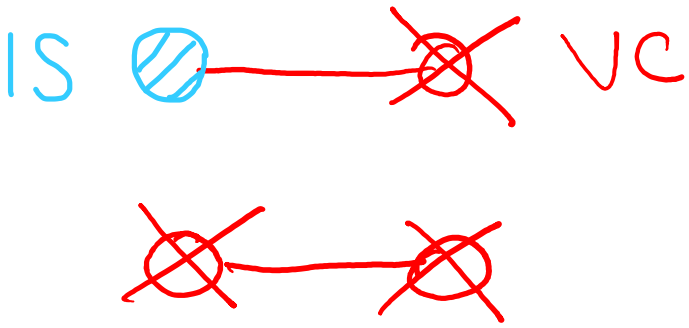
?

VC =^{c} Ind set



Independent set (IS): set of vertices that are not connected (white)

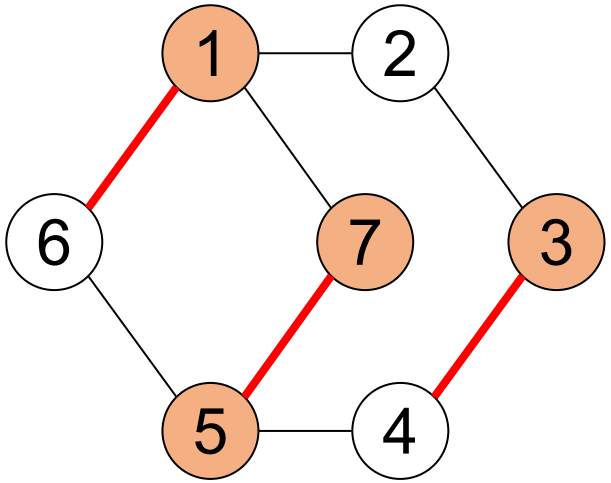
Vertex cover (VC): set of vertices that covers all edges (orange)



Set S is a **VC** iff the **complement** $V^c = V - S$ is an **IS**

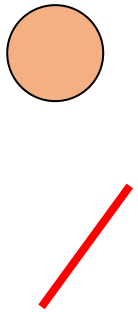
Proof: for each edge at most one vertex is in V^c.
Thus at least one vertex is in Set S.

Matching vs. VC?



Vertex cover (VC): set of vertices that covers all edges (orange)

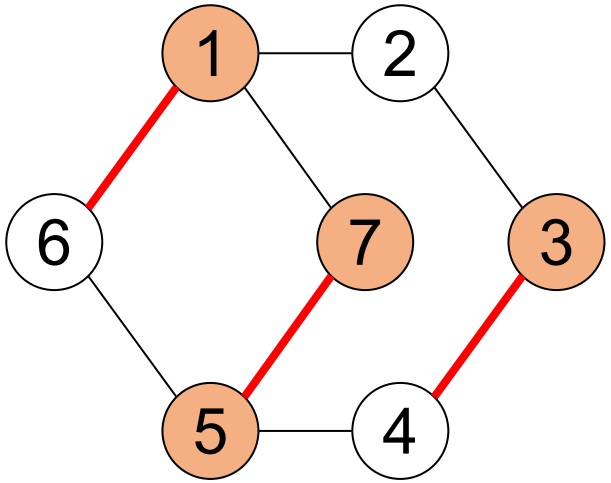
Matching (Ind edge set): set of edges w/o common vertices (red)



What is a possible connection between VC and matchings

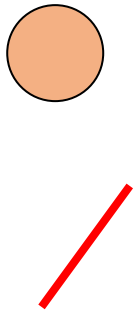
?

Matching ≤ VC



Vertex cover (VC): set of vertices that covers all edges (orange)

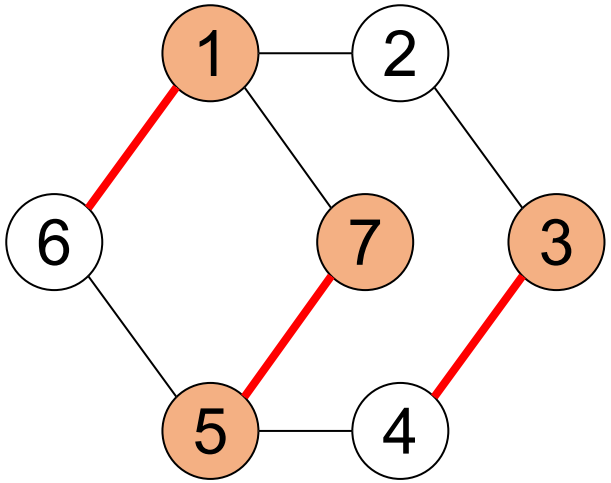
Matching (Ind edge set): set of edges w/o common vertices (red)



A **VC** needs to cover at least each edge from any matching

Thus, any VC has at least the size of any matching
⇒ **Size of any matching** ≤ any VC

Matching ≤ VC =^{c} Ind set (summary so far)



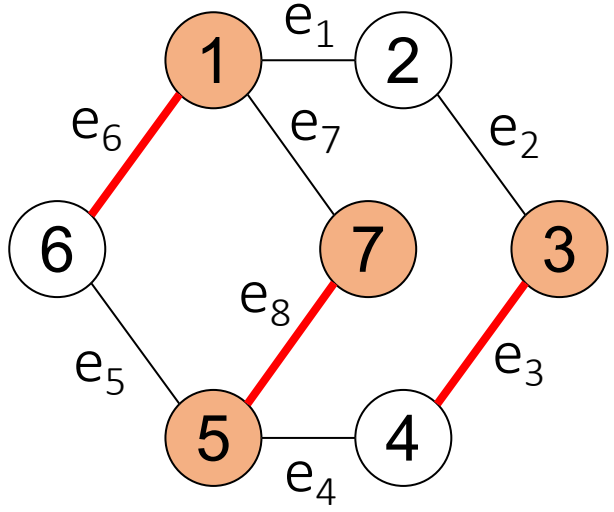
What intuitive problem is missing ?

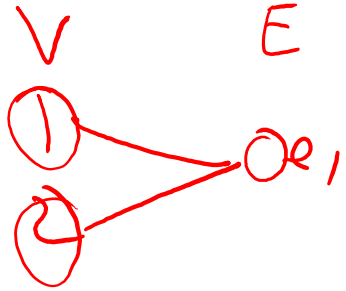
Independent set (IS): set of vertices that are not connected (white)

Vertex cover (VC): set of vertices that covers all edges (orange)

Matching (Ind edge set): set of edges w/o common vertices (red)

Matching ≤ VC =^{c} Ind set (summary so far)





What intuitive problem is missing ?

Independent set (IS): set of vertices that are not connected (white)

Vertex cover (VC): set of vertices that covers all edges (orange)

Matching (Ind edge set): set of edges w/o common vertices (red)

Edges = Sets

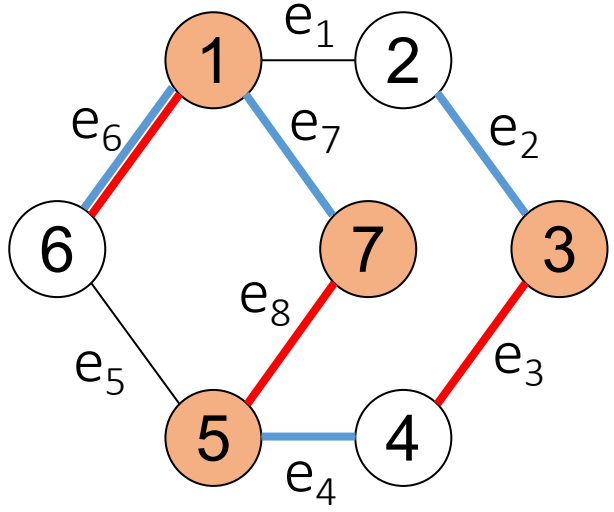
Vertices = elements

	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8
1	o					o	o	
2	o	o						
3		o	o					
4			o	o				
5				o	o			o
6					o	o		
7							o	o

Cover problems: set of subsets that cover all elements

Packing problems: set of disjoint subsets (max set

Matching ≤ VC =^{c} Ind set vs. Edge cover



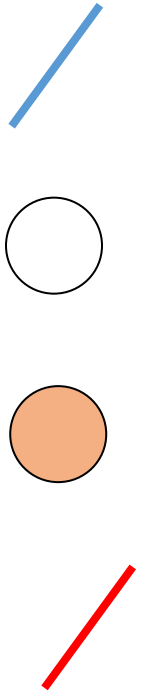
What is its connection of edge cover ?

Edge cover: set of edges that cover all vertices (blue)

Independent set (IS): set of vertices that are not connected (white)

Vertex cover (VC): set of vertices that covers all edges (orange)

Matching (Ind edge set): set of edges w/o common vertices (red)



Edges = Sets

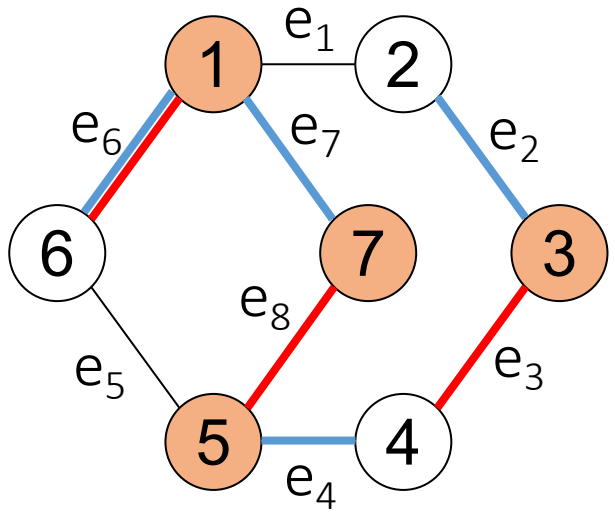
Vertices = elements

	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8
1	o					o	o	
2	o	o						
3		o	o					
4			o	o				
5				o	o			o
6					o	o		
7							o	o

Cover problems: set of subsets that cover all elements
(min set cover: min vertex cover, min edge cover)

Packing problems: set of disjoint subsets
(max set packing: max ind set, max matching)

Matching ≤ VC =^{c} Ind set ≤ Edge cover



Edges = Sets

Vertices = elements

	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8
1	o					o	o	
2	o	o						
3		o	o					
4			o	o				
5				o	o			o
6					o	o		
7							o	o

Edge cover: set of edges that cover all vertices (blue)

Independent set (IS): set of vertices that are not connected (white)

Vertex cover (VC): set of vertices that covers all edges (orange)

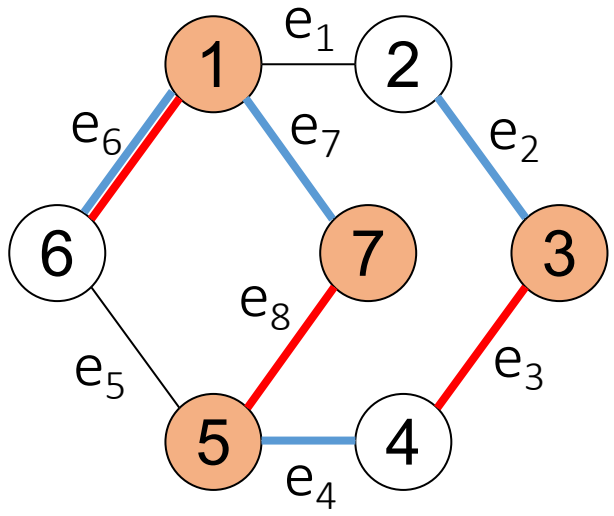
Matching (Ind edge set): set of edges w/o common vertices (red)

An **edge cover** needs to cover at least each vertex from any IS

Thus, any IS is lower bound to the size of any edge cover

⇒ **Size of min edge cover** ≥ max IS

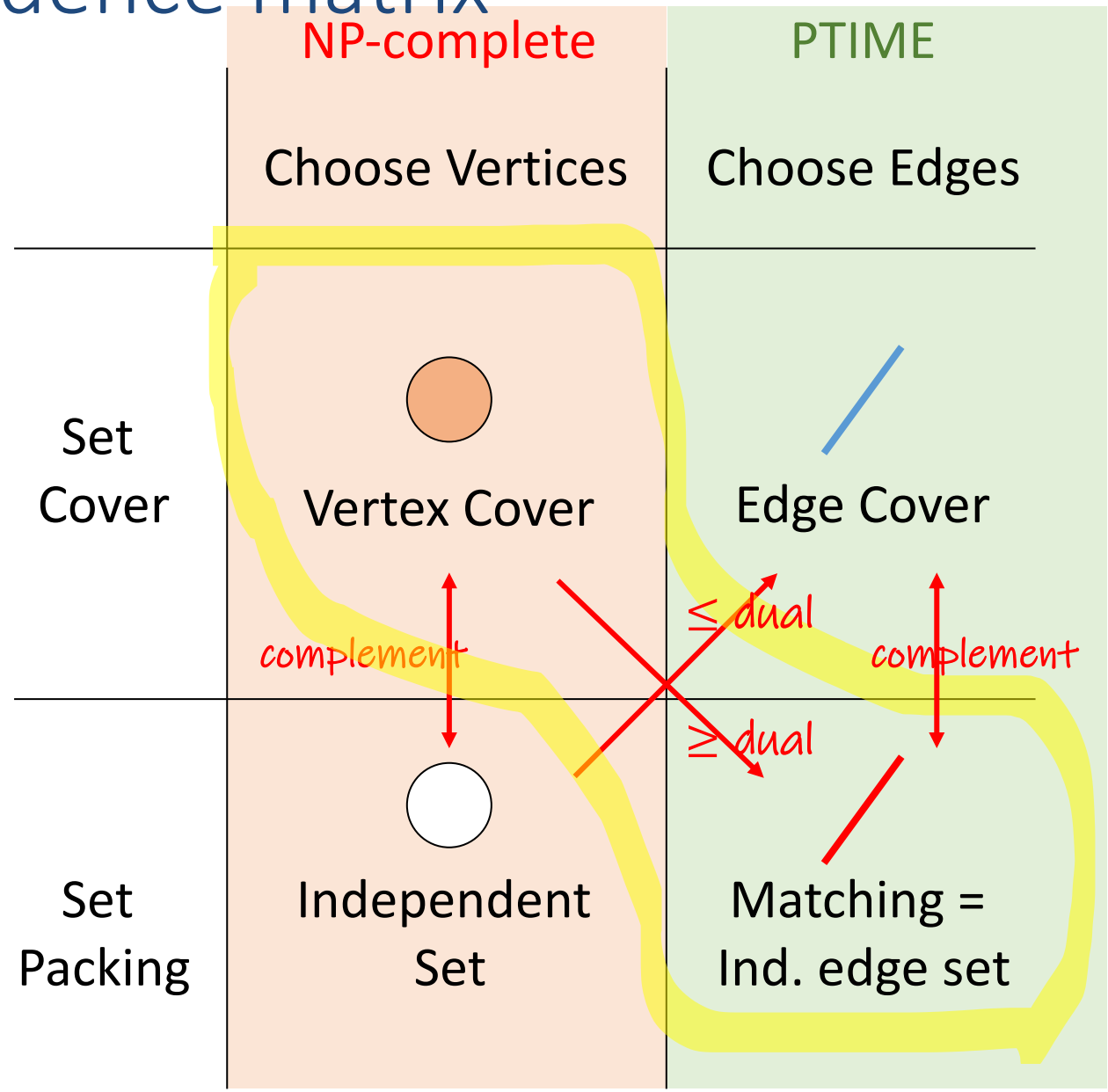
4 graph problems in the incidence matrix



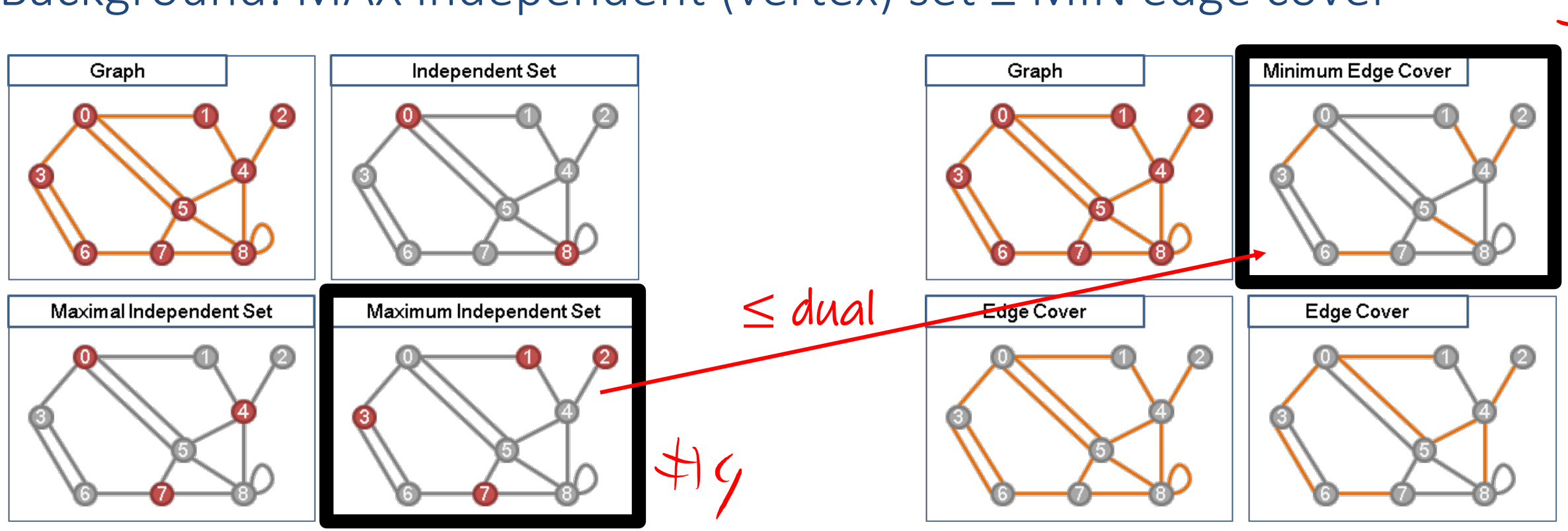
Edges = Sets

Vertices = elements

	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8
1	o					o	o	
2	o	o						
3		o	o					
4			o	o				
5				o	o			o
6					o	o		
7							o	o



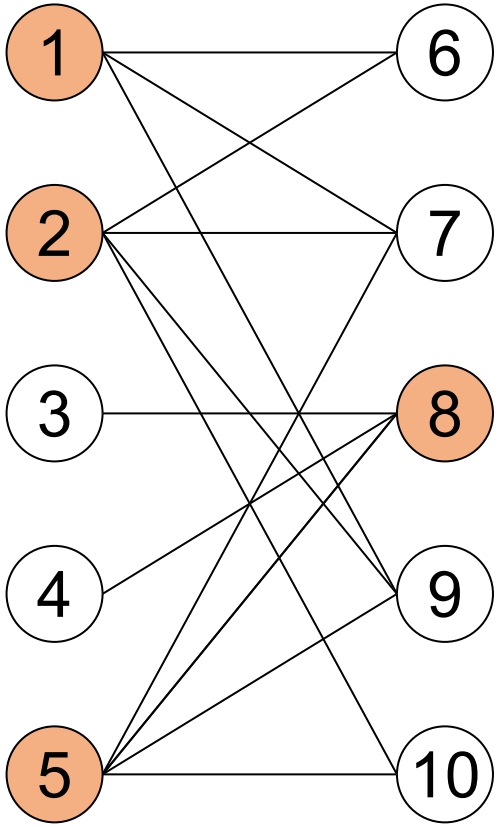
Background: MAX independent (vertex) set ≤ MIN edge cover



- Assume graph G is connected. Thus, every vertex has at least one edge (unless just one vertex)
- Suppose S is an independent set and E is an edge cover.
- Then for each vertex $v \in S$ there exists at least one edge $e \in E$ incident with v.
- By definition of independent set no two $u, v \in S$, have a common edge in E.
- Therefore $|S| \leq |E|$

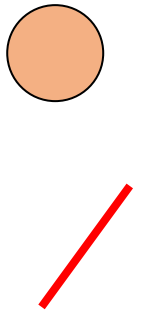
Examples from: http://www.csie.ntnu.edu.tw/~u91029/Domination.html

Matching ≤ VC: what changes in bipartite graphs?



Vertex cover (VC): set of vertices that covers all edges (orange)

Matching (Ind edge set): set of edges w/o common vertices (red)

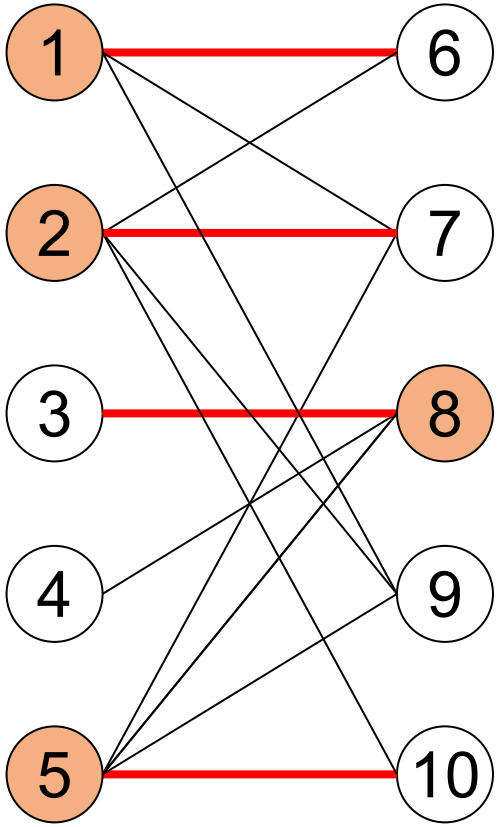


A **VC** needs to cover at least each edge from any matching

Thus, min VC at least the size of any matching

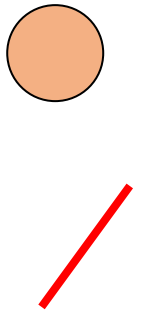
⇒ **Size of any matching** ≤ any VC

matching = VC ... in bipartite graphs!



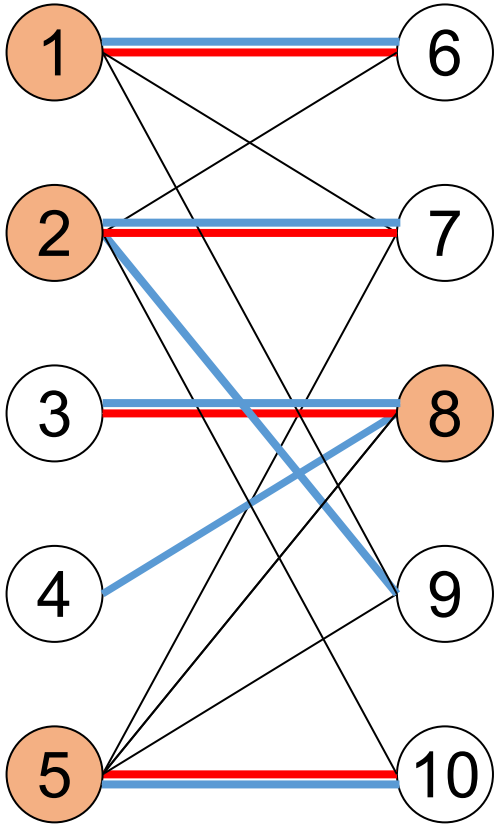
Vertex cover (VC): set of vertices that covers all edges (orange)

Matching (Ind edge set): set of edges w/o common vertices (red)



Kőnig-Egeváry theorem for bipartite graphs:
Max matching equivalent to Min VC

All for 4 problems become easy in bipartite graphs



PTIME

	Choose Vertices	Choose Edges
Set Cover	Vertex Cover	Edge Cover
Set Packing	Independent Set	Matching = Ind. edge set

complement

= dual

= dual

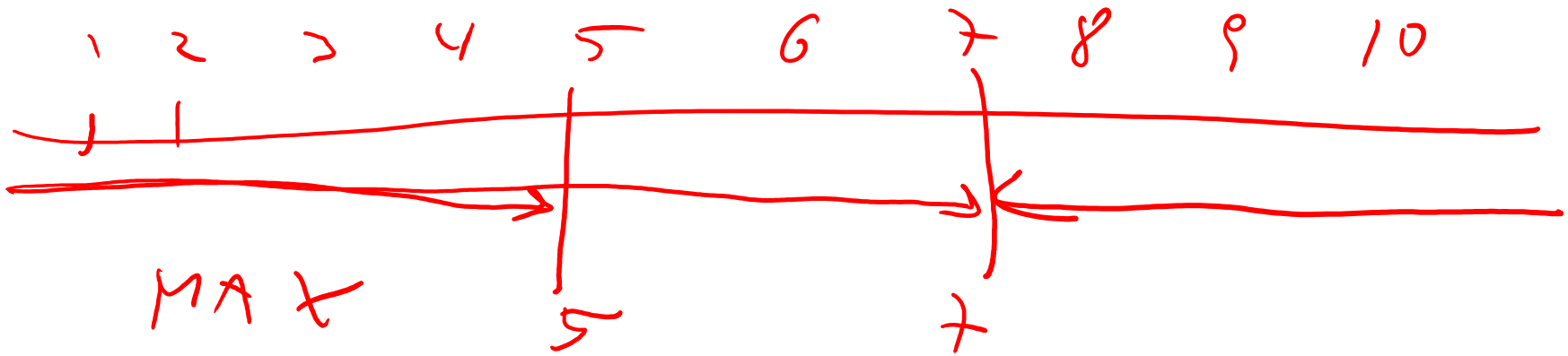
complement

LP (Linear Programming) and duality gaps

Dual Optimization Problem

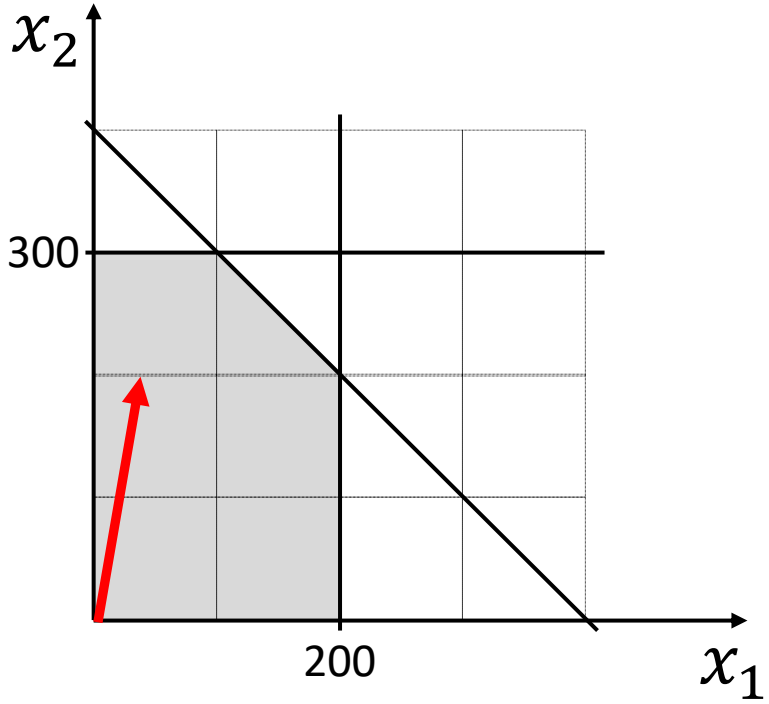
(e.g, max independent set) (e.g., min edge cover)

- A maximization problem **M** and a minimization problem **N**, defined on the same instances (such as graphs, constraints) s.t.:
  1. for every candidate solution *M* to **M** and every candidate solution *N* to **N**, the value of *M* is less than or equal to the value of *N*
  2. obtaining candidate solutions *M* and *N* that have the same value proves that *M* and *N* are optimal solutions for that instance.



A quick primer on Duality in Linear Programming

$$\begin{aligned} \max\ & x_1 + 6x_2 \\ & x_1 \leq 200 \\ & x_2 \leq 300 \\ & x_1 + x_2 \leq 400 \\ & x_1, x_2 \geq 0 \end{aligned}$$



optimum solution to be $(x_1, x_2) = (100, 300)$

objective value 1900

Example taken from: Dasgupta, Papadimitriou, Vazirani. Algorithms. 2006. http://algorithmics.lsi.upc.edu/docs/Dasgupta-Papadimitriou-Vazirani.pdf
Wolfgang Gatterbauer. Principles of scalable data management: https://northeastern-datalab.github.io/cs7240/

A quick primer on Duality in Linear Programming

$$\begin{aligned} \max\ & x_1 + 6x_2 \\ & x_1 \leq 200 \\ & x_2 \leq 300 \\ & x_1 + x_2 \leq 400 \\ & x_1, x_2 \geq 0 \end{aligned}$$

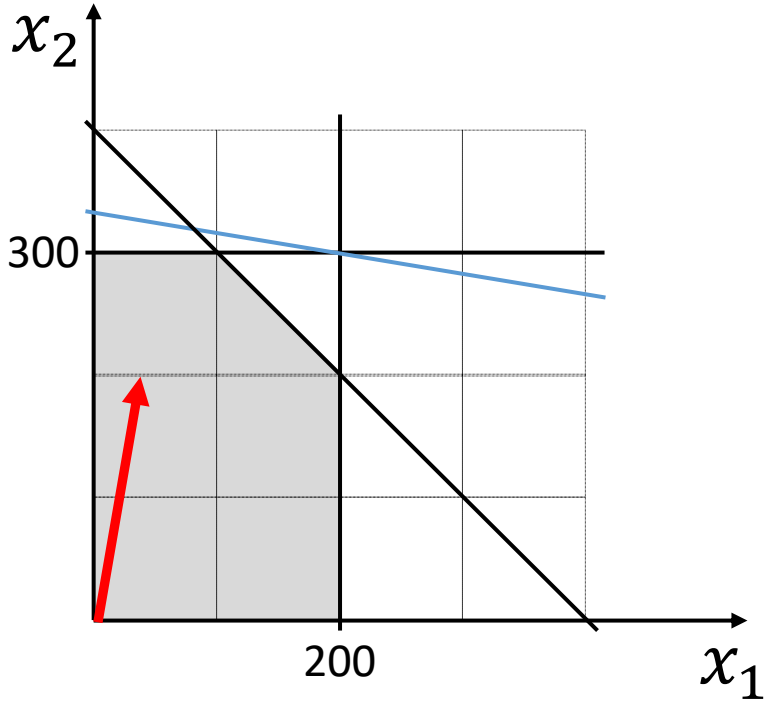
x 1

x 6

x 0

$$x_1 + 6x_2 \leq 2000$$

upper bound!



optimum solution to be $(x_1, x_2) = (100, 300)$

objective value 1900

Example taken from: Dasgupta, Papadimitriou, Vazirani. Algorithms. 2006. http://algorithmics.lsi.upc.edu/docs/Dasgupta-Papadimitriou-Vazirani.pdf
Wolfgang Gatterbauer. Principles of scalable data management: https://northeastern-datalab.github.io/cs7240/

A quick primer on Duality in Linear Programming

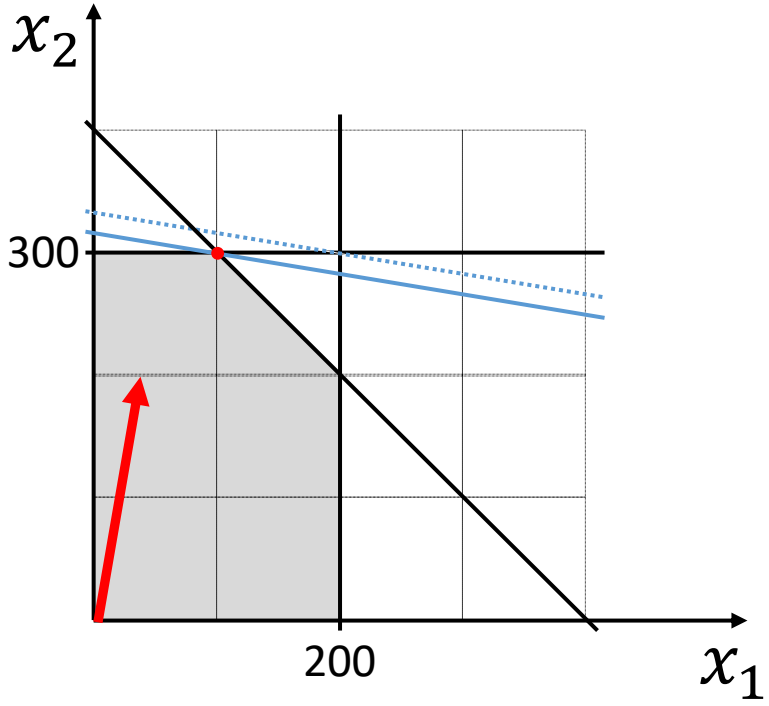
$$\begin{aligned} \max\ & x_1 + 6x_2 \\ & x_1 \leq 200 \\ & x_2 \leq 300 \\ & x_1 + x_2 \leq 400 \\ & x_1, x_2 \geq 0 \end{aligned}$$

x 0

x 5

x 1

$$x_1 + 6x_2 \leq 1900$$



(0,5,1)... certificate of optimiality

optimum solution to be $(x_1, x_2) = (100, 300)$

objective value 1900

Example taken from: Dasgupta, Papadimitriou, Vazirani. Algorithms. 2006. http://algorithmics.lsi.upc.edu/docs/Dasgupta-Papadimitriou-Vazirani.pdf
Wolfgang Gatterbauer. Principles of scalable data management: https://northeastern-datalab.github.io/cs7240/

A quick primer on Duality in Linear Programming

non-negative!

$$\begin{aligned} \max\ & x_1 + 6x_2 \\ & x_1 \leq 200 \\ & x_2 \leq 300 \\ & x_1 + x_2 \leq 400 \\ & x_1, x_2 \geq 0 \end{aligned}$$

Multiplier	Inequality
y_1 (x 0)	$x_1 \leq 200$
y_2 (x 5)	$x_2 \leq 300$
y_3 (x 1)	$x_1 + x_2 \leq 400$

$$x_1 + 6x_2 \leq 1900$$

(0,5,1)... certificate of optimiality

Example taken from: Dasgupta, Papadimitriou, Vazirani. Algorithms. 2006. http://algorithmics.lsi.upc.edu/docs/Dasgupta-Papadimitriou-Vazirani.pdf
Wolfgang Gatterbauer. Principles of scalable data management: https://northeastern-datalab.github.io/cs7240/

A quick primer on Duality in Linear Programming

non-negative!

$$\begin{aligned} \max\ & x_1 + 6x_2 \\ & x_1 \le 200 \\ & x_2 \le 300 \\ & x_1 + x_2 \le 400 \\ & x_1, x_2 \ge 0 \end{aligned}$$

Multiplier	Inequality
y_1	$x_1 \le 200$
y_2	$x_2 \le 300$
y_3	$x_1 + x_2 \le 400$

$$(y_1 + y_3)x_1 + (y_2 + y_3)x_2 \le \boxed{200y_1 + 300y_2 + 400y_3}$$

if ≥ 1 and ≥ 6 then right side upper bound for objective ($x_1 + 6x_2$)

Example taken from: Dasgupta, Papadimitriou, Vazirani. Algorithms. 2006. http://algorithmics.lsi.upc.edu/docs/Dasgupta-Papadimitriou-Vazirani.pdf
Wolfgang Gatterbauer. Principles of scalable data management: https://northeastern-datalab.github.io/cs7240/

A quick primer on Duality in Linear Programming

$$\textbf{Primal}: (x_1, x_2) = (100, 300); \quad \textbf{Dual}: (y_1, y_2, y_3) = (0, 5, 1)$$

$$\begin{aligned} \max\ & x_1 + 6x_2 \\ & x_1 \le 200 \\ & x_2 \le 300 \\ & x_1 + x_2 \le 400 \\ & x_1, x_2 \ge 0 \end{aligned}$$

$$\begin{aligned} \min\ & 200y_1 + 300y_2 + 400y_3 \\ & y_1 + y_3 \ge 1 \\ & y_2 + y_3 \ge 6 \\ & y_1, y_2, y_3 \ge 0 \end{aligned}$$

$$(y_1 + y_3)x_1 + (y_2 + y_3)x_2 \le 200y_1 + 300y_2 + 400y_3$$

if ≥ 1 and ≥ 6

then right side upper bound for objective $(x_1 + 6x_2)$

Example taken from: Dasgupta, Papadimitriou, Vazirani. Algorithms. 2006. http://algorithmics.lsi.upc.edu/docs/Dasgupta-Papadimitriou-Vazirani.pdf
Wolfgang Gatterbauer. Principles of scalable data management: https://northeastern-datalab.github.io/cs7240/

A quick primer on Duality in Linear Programming

Figure 7.10 A generic primal LP in matrix-vector form, and its dual.

Primal LP:

$$\max \ \mathbf{c}^T\mathbf{x}$$
$$\mathbf{Ax} \leq \mathbf{b}$$
$$\mathbf{x} \geq 0$$

Dual LP:

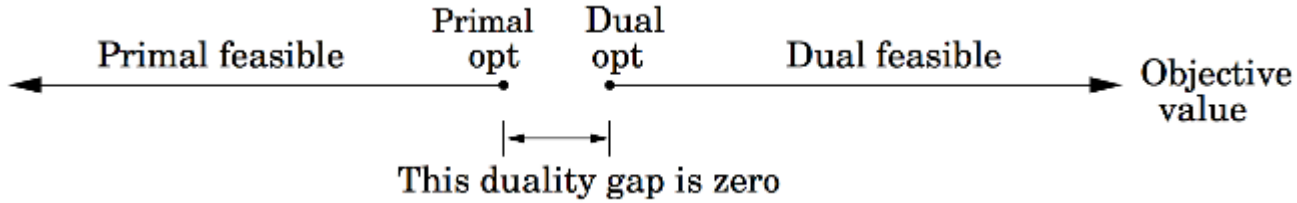
$$\min \ \mathbf{y}^T\mathbf{b}$$
$$\mathbf{y}^T\mathbf{A} \geq \mathbf{c}^T$$
$$\mathbf{y} \geq 0$$

Primal LP:

$$\max \ c_1x_1 + \cdots + c_nx_n$$
$$a_{i1}x_1 + \cdots + a_{in}x_n \leq b_i \quad \text{for } i \in I$$
$$a_{i1}x_1 + \cdots + a_{in}x_n = b_i \quad \text{for } i \in E$$
$$x_j \geq 0 \quad \text{for } j \in N$$

Dual LP:

$$\min \ b_1y_1 + \cdots + b_my_m$$
$$a_{1j}y_1 + \cdots + a_{mj}y_m \geq c_j \quad \text{for } j \in N$$
$$a_{1j}y_1 + \cdots + a_{mj}y_m = c_j \quad \text{for } j \notin N$$
$$y_i \geq 0 \quad \text{for } i \in I$$



Example taken from: Dasgupta, Papadimitriou, Vazirani. Algorithms. 2006. http://algorithmics.lsi.upc.edu/docs/Dasgupta-Papadimitriou-Vazirani.pdf
Wolfgang Gatterbauer. Principles of scalable data management: https://northeastern-datalab.github.io/cs7240/

LP relaxations of ILP problems (Integer Linear Programming)

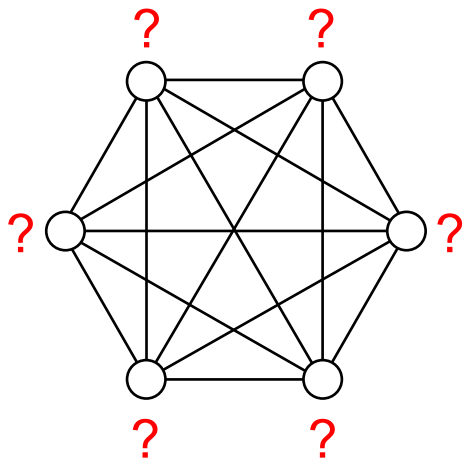
Example: Minimal (Fractional) Vertex Cover in k-clique

Objective: $\min \sum_{v \in V} w_v$ s.t. $w_v + w_u \geq 1$ for each edge

and $0 \leq w_v \leq 1$ for each node for integral solution (ILP)

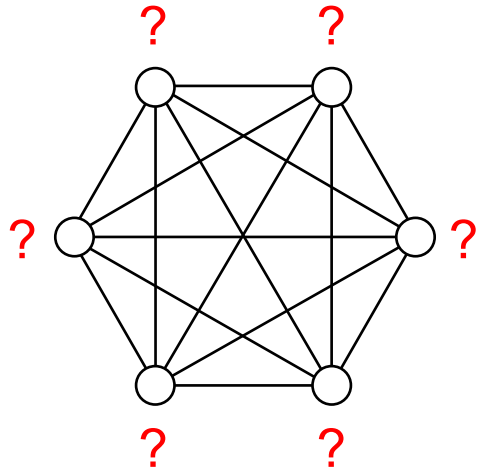
or $w_v \in \{0,1\}$ for each node for fractional solution (LP)

Minimal Integral Vertex Cover:



ILP: ?

Minimal Fractional Vertex Cover:



LP: ?

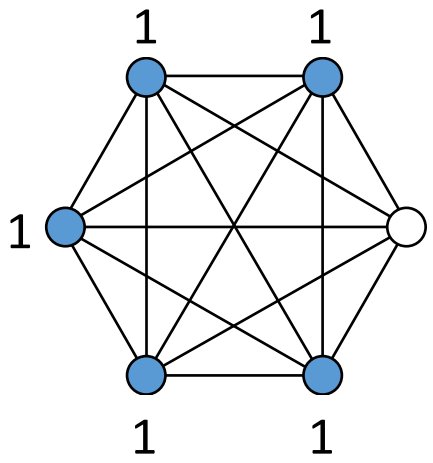
Example: Minimal (Fractional) Vertex Cover in k-clique

Objective: $\min \sum_{v \in V} w_v$ s.t. $w_v + w_u \geq 1$ for each edge

and $0 \leq w_v \leq 1$ for each node for integral solution (ILP)

or $w_v \in \{0,1\}$ for each node for fractional solution (LP)

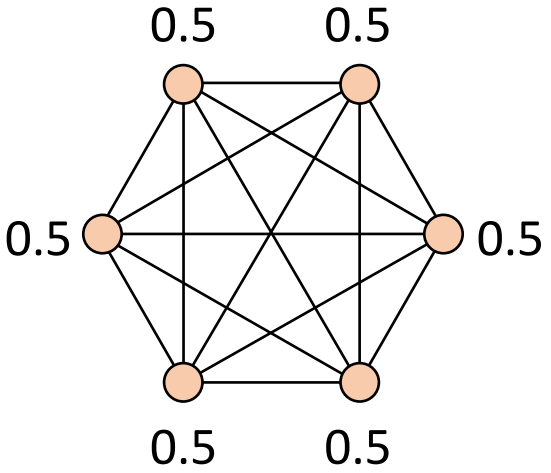
Minimal Integral Vertex Cover:



ILP: 5 = k-1

for k-clique

Minimal Fractional Vertex Cover:



LP: 3 = k/2

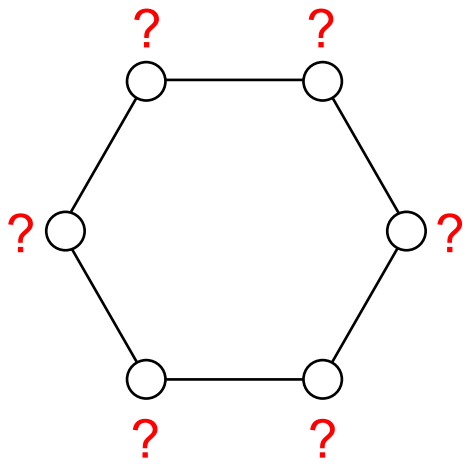
Example: Minimal (Fractional) Vertex Cover in k-clique

Objective: $\min \sum_{v \in V} w_v$ s.t. $w_v + w_u \geq 1$ for each edge

and $0 \leq w_v \leq 1$ for each node for integral solution (ILP)

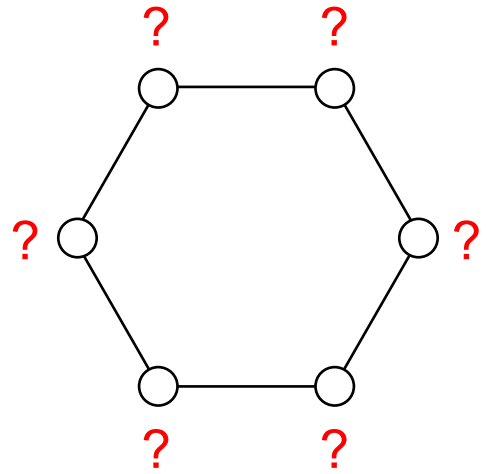
or $w_v \in \{0,1\}$ for each node for fractional solution (LP)

Minimal Integral Vertex Cover:



ILP: ?

Minimal Fractional Vertex Cover:



LP: ?

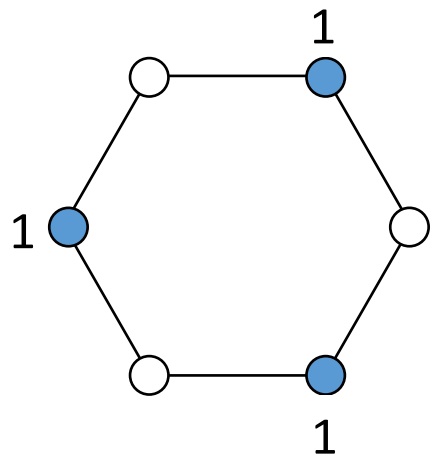
Example: Minimal (Fractional) Vertex Cover in k-clique

Objective: $\min \sum_{v \in V} w_v$ s.t. $w_v + w_u \geq 1$ for each edge

and $0 \leq w_v \leq 1$ for each node for integral solution (ILP)

or $w_v \in \{0,1\}$ for each node for fractional solution (LP)

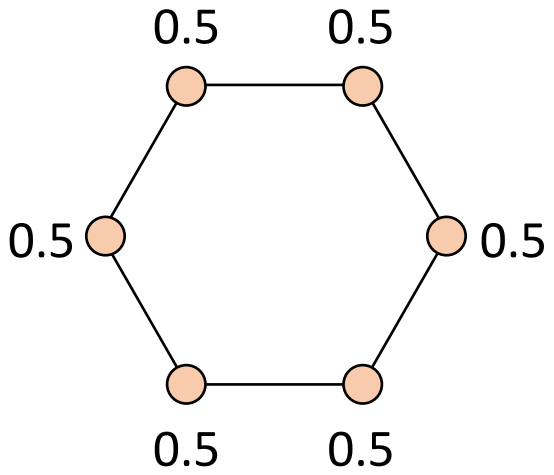
Minimal Integral Vertex Cover:



ILP: 3 = k/2

for even cycle

Minimal Fractional Vertex Cover:



LP: 3 = k/2

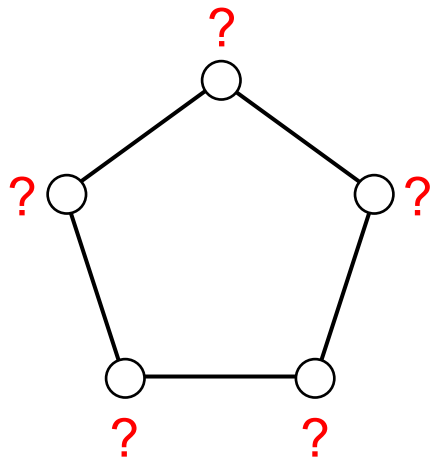
Example: Minimal (Fractional) Vertex Cover in k-clique

Objective: $\min \sum_{v \in V} w_v$ s.t. $w_v + w_u \geq 1$ for each edge

and $0 \leq w_v \leq 1$ for each node for integral solution (ILP)

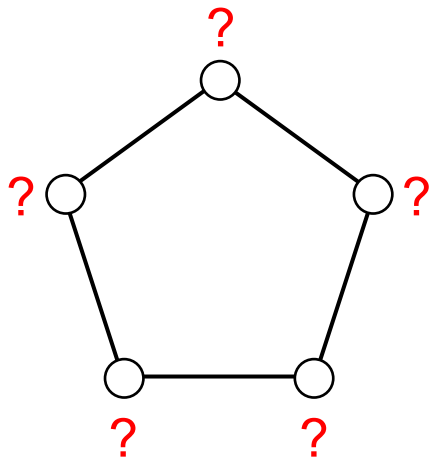
or $w_v \in \{0,1\}$ for each node for fractional solution (LP)

Minimal Integral Vertex Cover:



ILP: ?

Minimal Fractional Vertex Cover:



LP: ?

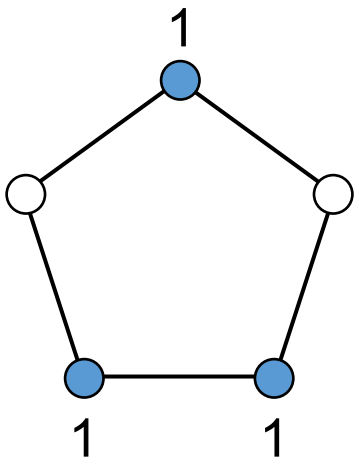
Example: Minimal (Fractional) Vertex Cover in k-clique

Objective: $\min \sum_{v \in V} w_v$ s.t. $w_v + w_u \geq 1$ for each edge

and $0 \leq w_v \leq 1$ for each node for integral solution (ILP)

or $w_v \in \{0,1\}$ for each node for fractional solution (LP)

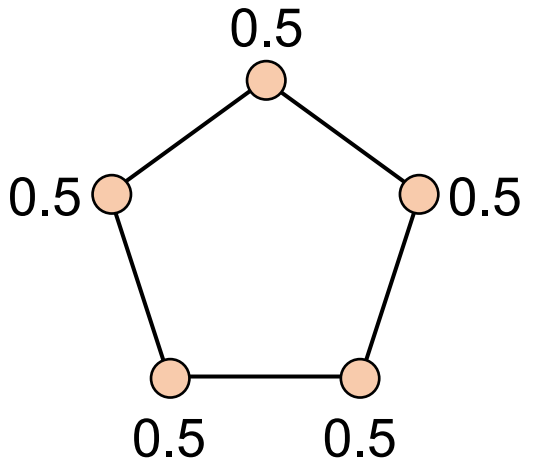
Minimal Integral Vertex Cover:



ILP: 3 = (k+1)/2

for odd cycle

Minimal Fractional Vertex Cover:

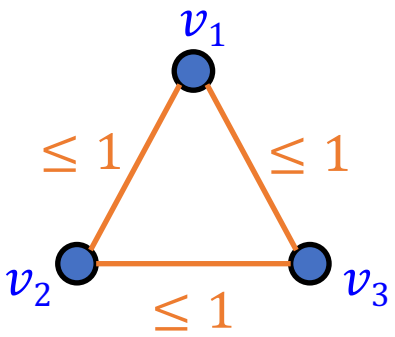


LP: 2.5 = k/2

Duality b/w independent vertex sets and edge covers

A quick primer on Duality in Linear Programming

Primal: Independence (Vertex) set



non-negative multiplier per edge

$$\max v_1 + v_2 + v_3 \text{ , s.t.}$$

$v_1 + v_2 \leq 1$	u_1	$v_1 + v_2 \leq 1$
$v_1 + v_3 \leq 1$	u_2	$v_1 + v_3 \leq 1$
$v_2 + v_3 \leq 1$	u_3	$v_2 + v_3 \leq 1$

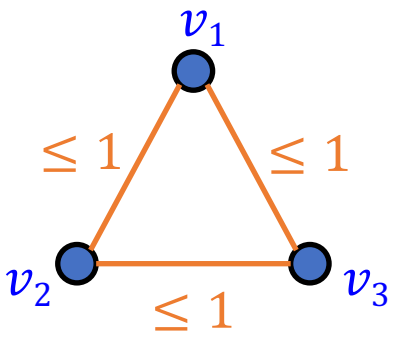
$$(u_1 + u_2)v_1 + (u_1 + u_3)v_2 + (u_2 + u_3)v_3 \leq u_1 + u_2 + u_3$$

if ≥ 1 ≥ 1 ≥ 1 for each vertex

then the right side $\sum_j u_j$ is an upper bound for the primal objective $\sum_i v_i$

A quick primer on Duality in Linear Programming

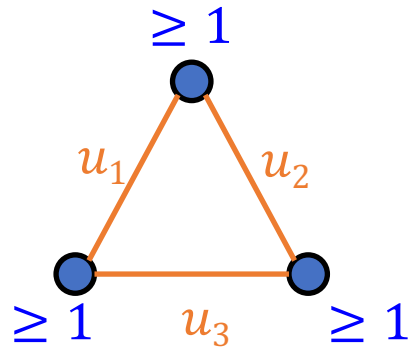
Primal: Independence (Vertex) set



$$\max\ v_1 + v_2 + v_3 \text{ , s.t.}$$
$$v_1 + v_2 \leq 1$$
$$v_1 + v_3 \leq 1$$
$$v_2 + v_3 \leq 1$$

Dual: Edge cover

$$\min\ u_1 + u_2 + u_3 \text{ , s.t.}$$
$$u_1 + u_2 \geq 1$$
$$u_1 + u_3 \geq 1$$
$$u_2 + u_3 \geq 1$$



$$(u_1 + u_2)v_1 + (u_1 + u_3)v_2 + (u_2 + u_3)v_3 \leq u_1 + u_2 + u_3$$

if ≥ 1 ≥ 1 ≥ 1

for each vertex

then the right side $\sum_j u_j$ is an upper bound for the primal objective $\sum_i v_i$

Example taken from: Dasgupta, Papadimitriou, Vazirani. Algorithms. 2006. http://algorithmics.lsi.upc.edu/docs/Dasgupta-Papadimitriou-Vazirani.pdf
Wolfgang Gatterbauer. Principles of scalable data management: https://northeastern-datalab.github.io/cs7240/

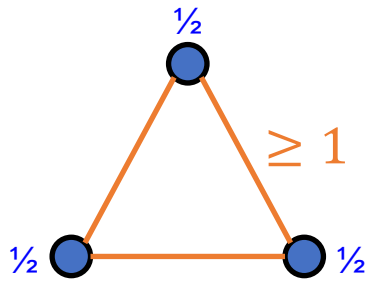
Independent Sets & Edge covers in the Triangle

Fractional vertex cover(τ^*)

min sum of weights $v_1, v_2, \ldots\ v_k \geq 0$ on vertices (variables)

s.t. for all $E(i,j)$: $v_i + v_j \geq 1$

$$\tau^* = \min \sum_i v_i$$



$$\min\ v_1 + v_2 + v_3 \text{ , s.t.}$$
$$v_1 + v_2 \geq 1$$
$$v_1 + v_3 \geq 1$$
$$v_2 + v_3 \geq 1$$

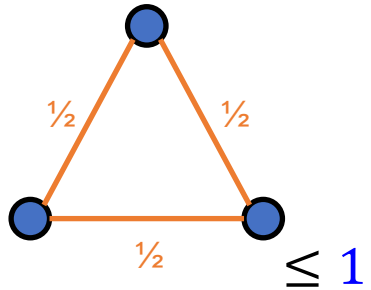
$$\tau^* = \nu^*$$

Fractional matching (edge packing) (ν^*)

max sum of weights $u_1, u_2, \ldots\ u_\ell \geq 0$ on edges (relations)

s.t. for all x_i: $\sum_{j: x_i \in R_j} u_j \leq 1$

$$\nu^* = \max \sum_j u_j$$



$$\min\ u_1 + u_2 + u_3 \text{ , s.t.}$$
$$u_1 + u_2 \leq 1$$
$$u_1 + u_3 \leq 1$$
$$u_2 + u_3 \leq 1$$

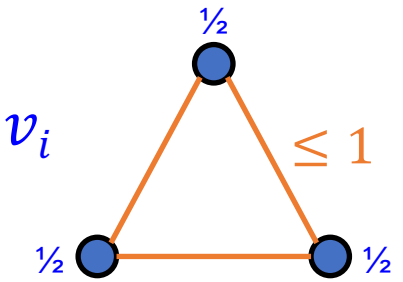
Independent Sets & Edge covers in the Triangle

Fractional independence number (α^*)

max sum of weights $v_1, v_2, \ldots\ v_k \geq 0$ on vertices (variables)

s.t. for all $E(i,j)$: $v_i + v_j \leq 1$

$$\alpha^* = \max \sum_i v_i$$



$$\max\ v_1 + v_2 + v_3 \text{ , s.t.}$$
$$v_1 + v_2 \leq 1$$
$$v_1 + v_3 \leq 1$$
$$v_2 + v_3 \leq 1$$

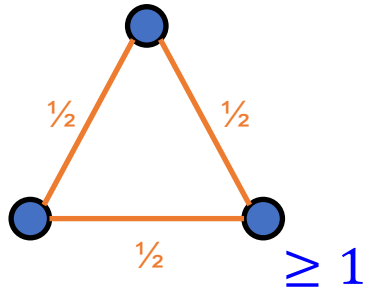
$$\alpha^* = \rho^*$$

Fractional edge cover (ρ^*)

min sum of weights $u_1, u_2, \ldots\ u_\ell \geq 0$ on edges (relations)

s.t. for all x_i: $\sum_{j: x_i \in E_j} u_j \geq 1$

$$\rho^* = \min \sum_j u_j$$

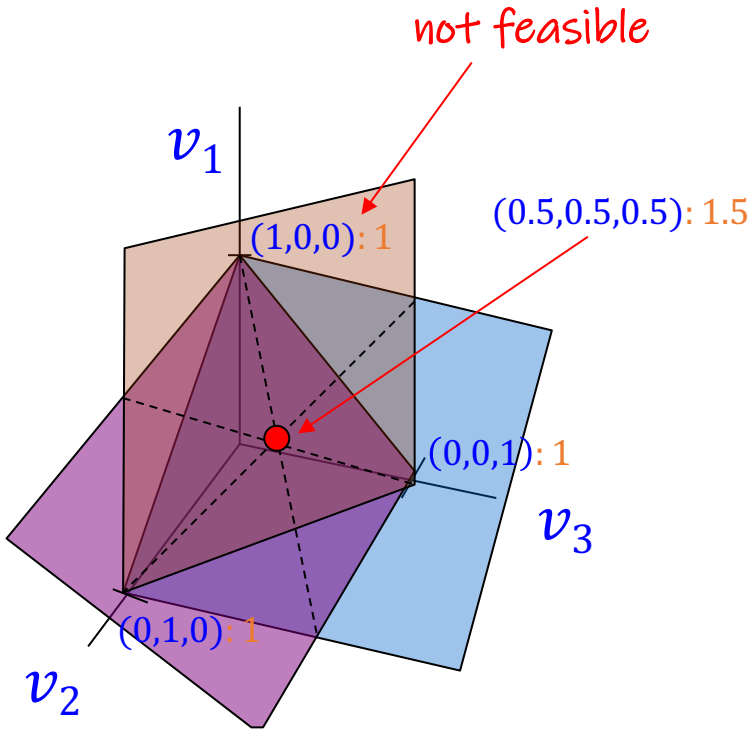


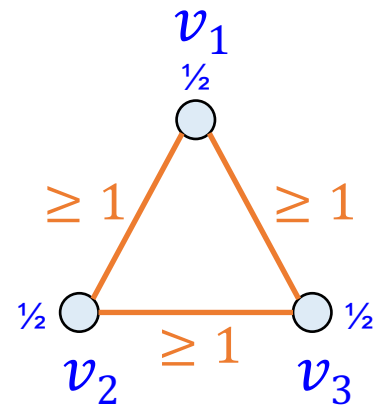
$$\min\ u_1 + u_2 + u_3 \text{ , s.t.}$$
$$u_1 + u_2 \geq 1$$
$$u_1 + u_3 \geq 1$$
$$u_2 + u_3 \geq 1$$

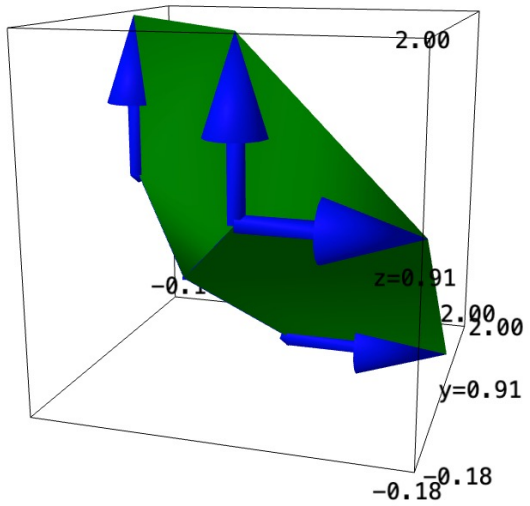
Fractional vertex cover in the triangle

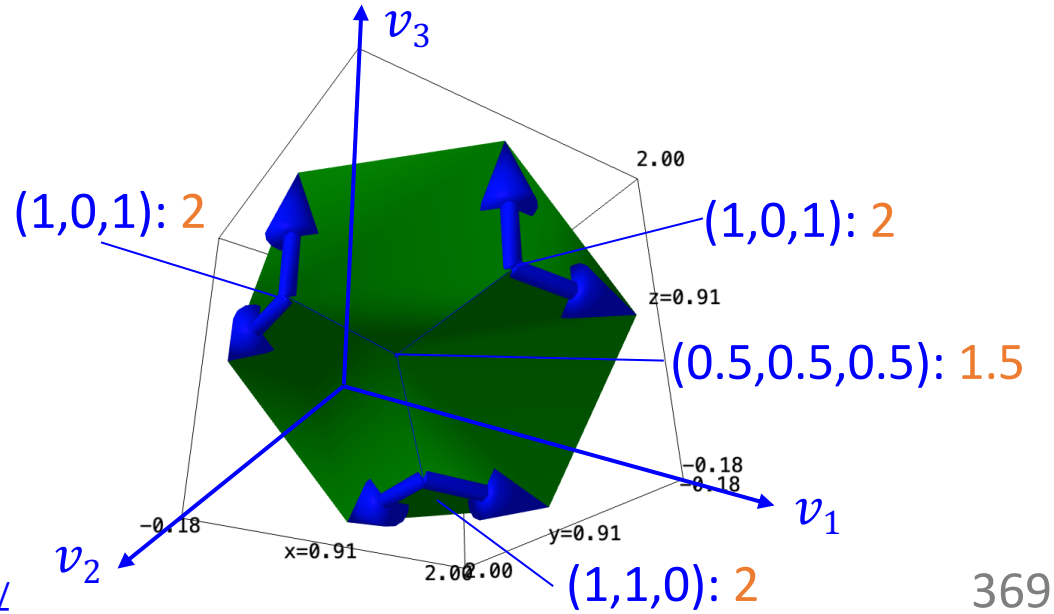
```
# https://sagecell.sagemath.org/
# inequalities: -1+v1+v2>=0, -1+v2+v3>=0, -1+v1+v3>=0, 1-v1>=0, 1-v2>=0, 1-v3>=0
p = Polyhedron(ieqs = [[-1,1,1,0],[-1,0,1,1],[-1,1,0,1],[0,1,0,0],[0,0,1,0],[0,0,0,1]])
p.plot()
```

$$\min v_1 + v_2 + v_3 \text{ , s.t.}$$
$$v_1 + v_2 \geq 1$$
$$v_1 + v_3 \geq 1$$
$$v_2 + v_3 \geq 1$$





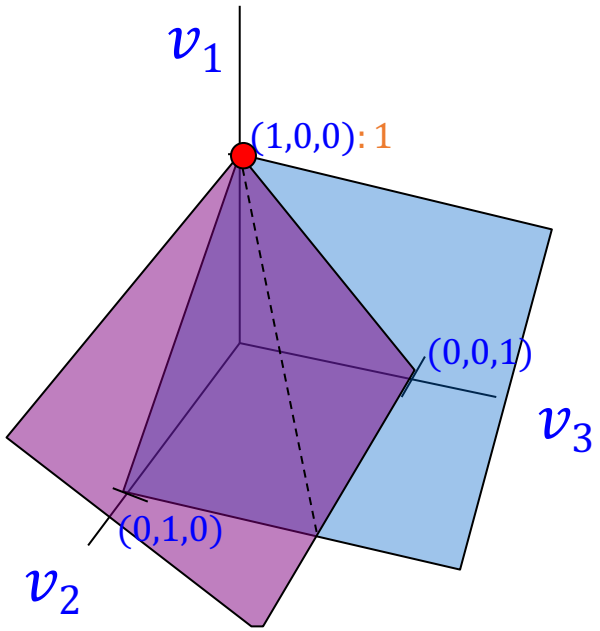


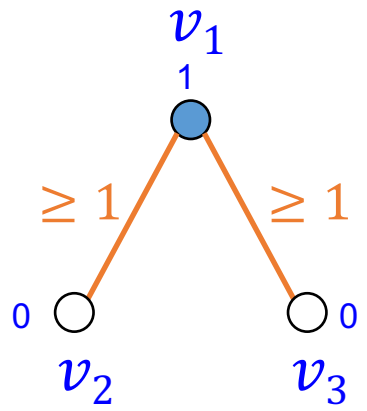


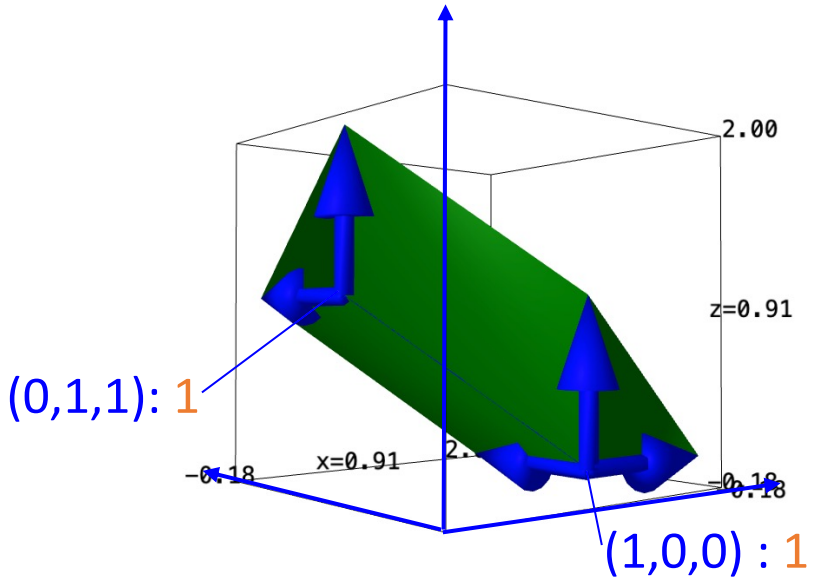
Fractional vertex cover in bipartite graph

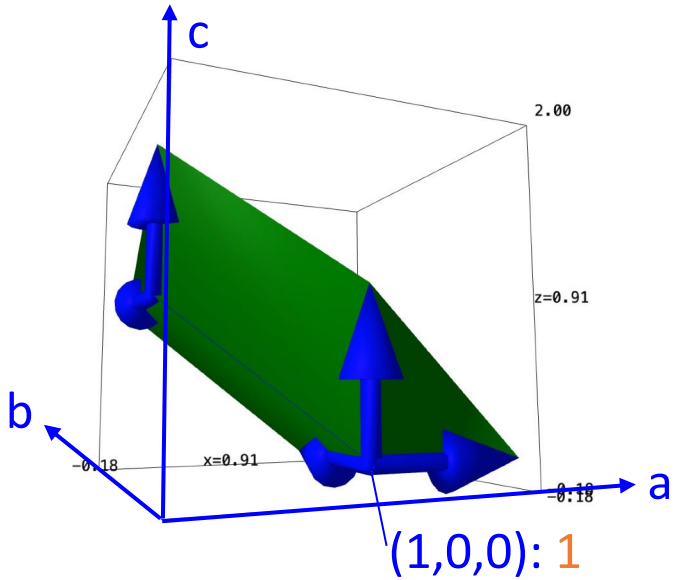
```
# https://sagecell.sagemath.org/
# inequalities: -1+v1+v2>=0, -1+v2+v3>=0, 1-v1>=0, 1-v2>=0, 1-v3>=0
p = Polyhedron(ieqs = [[-1,1,1,0],[-1,0,1,1],[0,1,0,0],[0,0,1,0],[0,0,0,1]])
p.plot()
```

$$\min v_1 + v_2 + v_3 \text{ , s.t.}$$
$$v_1 + v_2 \geq 1$$
$$v_1 + v_3 \geq 1$$









Outline: T3-2: Cyclic conjunctive queries

- T3-1: Acyclic conjunctive queries
- T3-2: Cyclic conjunctive queries
  - 2SAT (a detour)
  - Tree decompositions
  - Decompositions of hypertrees
  - Duality in Linear programming (a quick primer)
  - AGM bound (maximal result size for full CQs)
  - Worst-case optimal joins & the triangle query
  - Worst-case optimal joins & the 4-cycle
  - Optimal joins & the 4-cycle

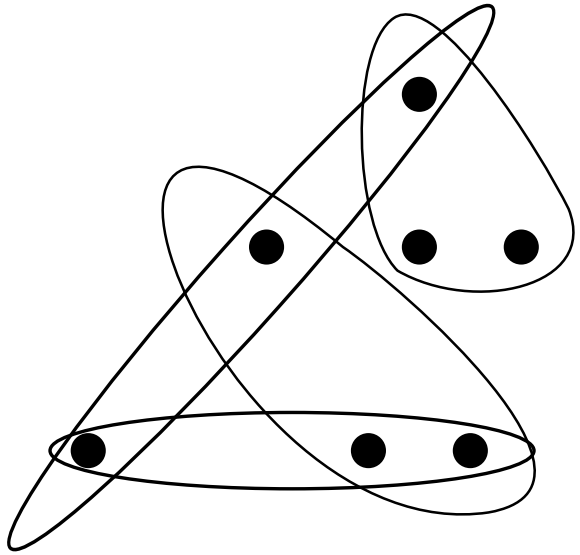
What do we know about bounding the size of the answer?

(. . .and enumerating all solutions)

Example by: Marx. "Graphs, hypergraphs, and the complexity of conjunctive database queries", ICDT 2017. http://edbticdt2017.unive.it/marx-icdt2017-talk.pdf
Wolfgang Gatterbauer. Principles of scalable data management: https://northeastern-datalab.github.io/cs7240/

Upper bound

Observation: If the hypergraph has edge cover number ρ and every relation has size at most N, then there are at most N^{ρ} tuples in the answer.

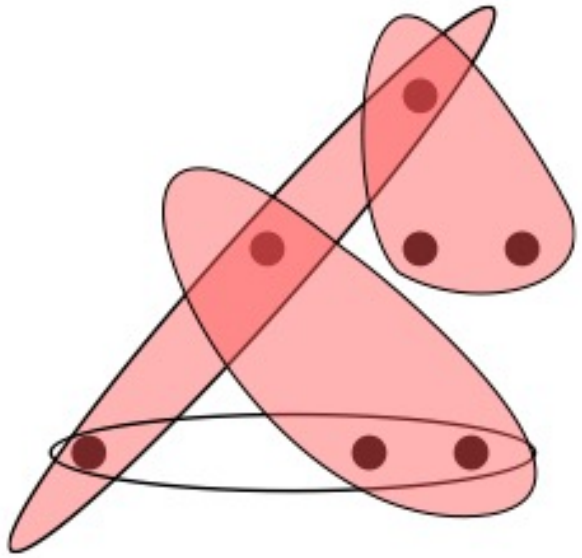


Example by: Marx. "Graphs, hypergraphs, and the complexity of conjunctive database queries", ICDT 2017. http://edbticdt2017.unive.it/marx-icdt2017-talk.pdf

Upper bound

minimal edge cover

Observation: If the hypergraph has edge cover number ρ and every relation has size at most N, then there are at most N^ρ tuples in the answer.



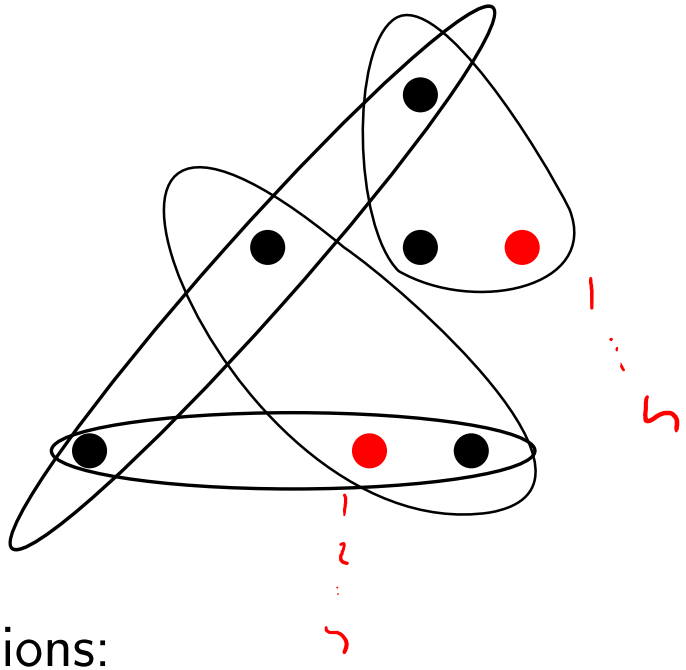
N^3

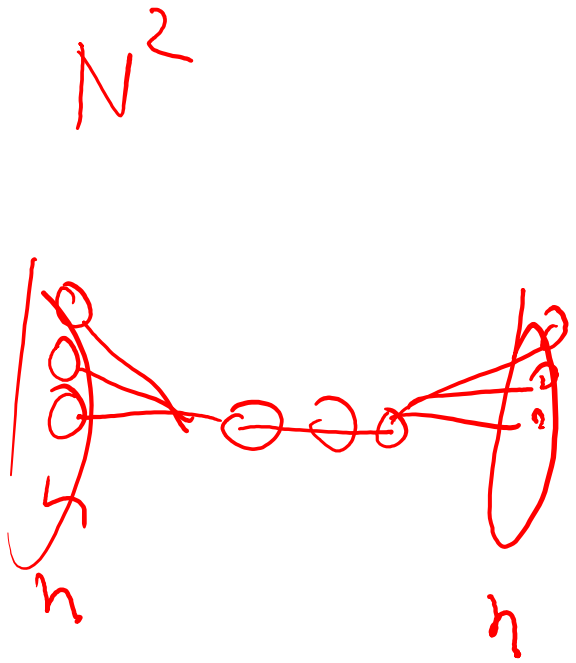
Example by: Marx. "Graphs, hypergraphs, and the complexity of conjunctive database queries", ICDT 2017. http://edbticdt2017.unive.it/marx-icdt2017-talk.pdf
Wolfgang Gatterbauer. Principles of scalable data management: https://northeastern-datalab.github.io/cs7240/

Lower bound

maximal independent set

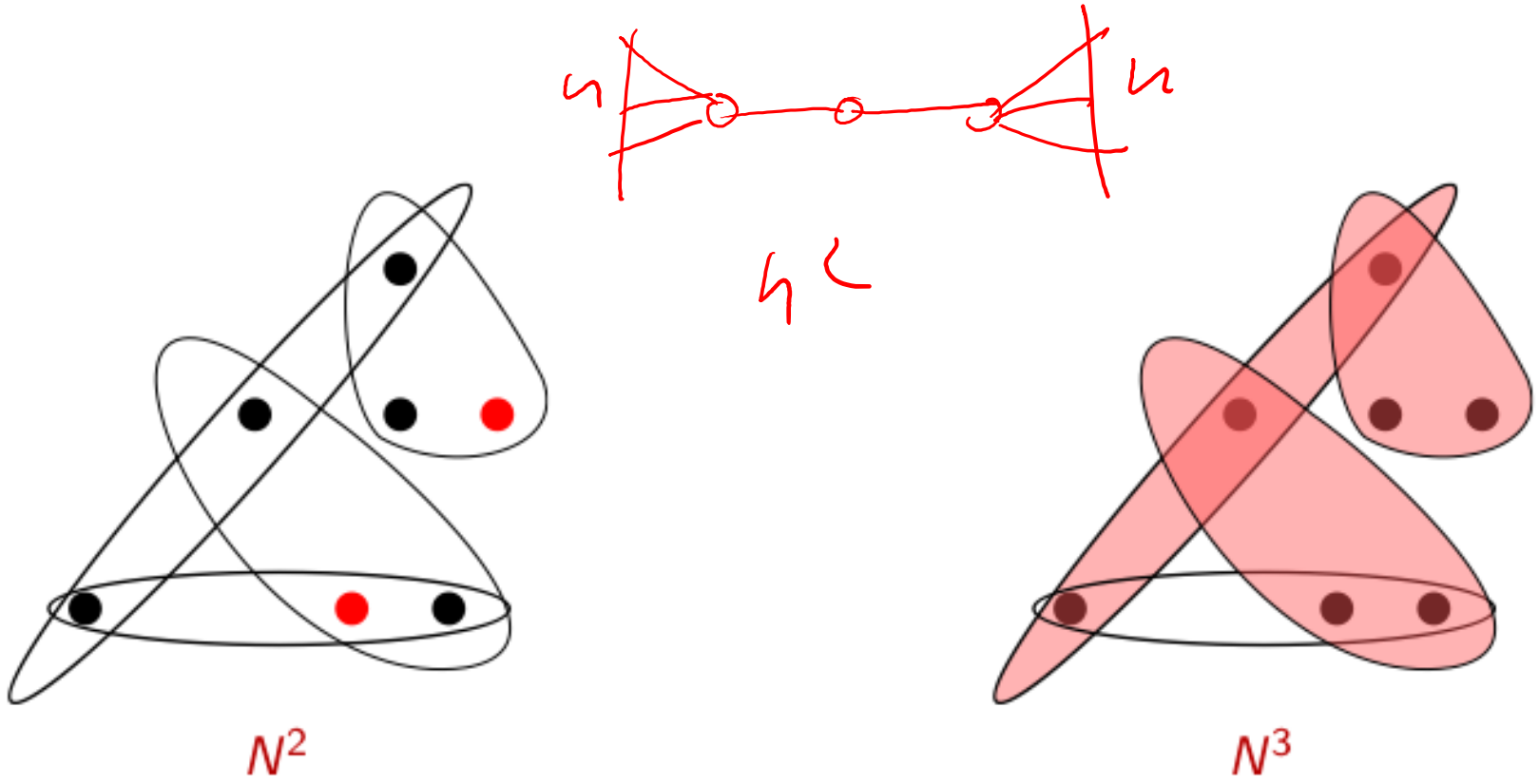
Observation: If the hypergraph has independence number α, then one can construct an instance where every relation has size N at the answer has size N^{α}.





Definition of the relations:

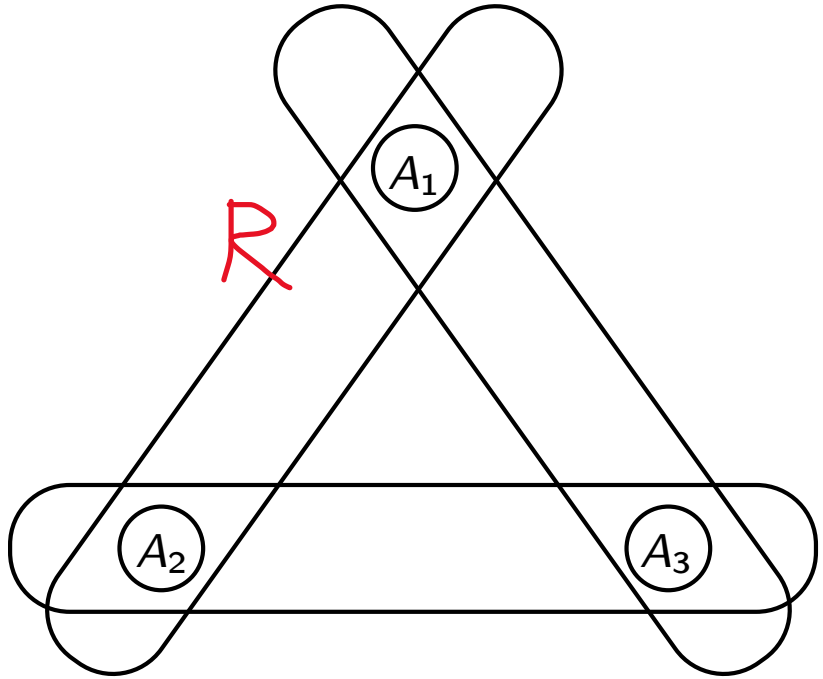
- If variable A is in the independent set, then it can take any value in $[N]$.
- Otherwise it is forced to 1.



Which is tight: the upper bound or the lower bound?

Example by: Marx. "Graphs, hypergraphs, and the complexity of conjunctive database queries", ICDT 2017. http://edbticdt2017.unive.it/marx-icdt2017-talk.pdf
Wolfgang Gatterbauer. Principles of scalable data management: https://northeastern-datalab.github.io/cs7240/

Example: triangles



Upper bound

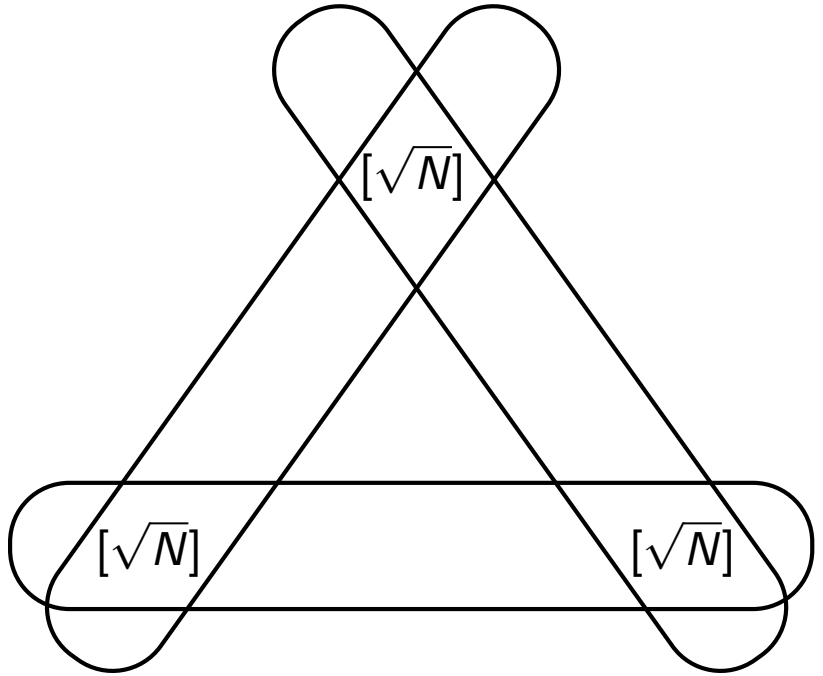
Two kind of values for A_1: in R

- Light: can be extended to at most $\sqrt{N}$ ways to A_2.
  $\Rightarrow \leq N \cdot \sqrt{N}$ answers with light A_1
- Heavy: can be extended to at least $\sqrt{N}$ ways to A_2.
  $\Rightarrow \leq \sqrt{N}$ heavy values $\Rightarrow \leq \sqrt{N} \cdot N$ answers with heavy A_1

$\Rightarrow$ At most $2 \cdot N^{3/2}$ answers.

Example by: Marx. "Graphs, hypergraphs, and the complexity of conjunctive database queries", ICDT 2017. http://edbticdt2017.unive.it/marx-icdt2017-talk.pdf

Example: triangles



Lower bound

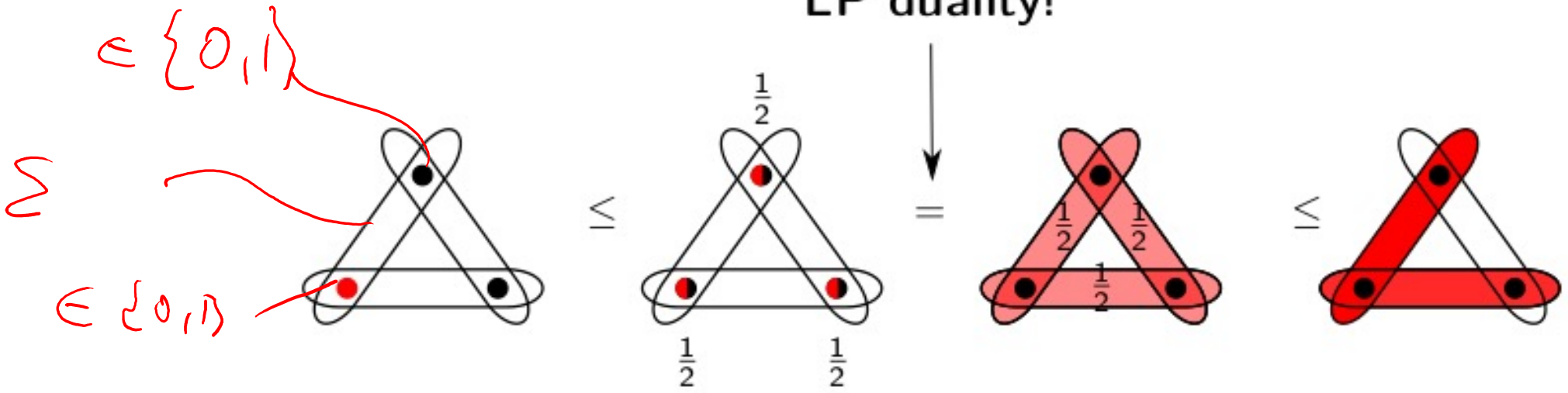
Allow every variable to be any value from $[\sqrt{N}] \Rightarrow N^{3/2}$ answers.

The correct bound $N^{3/2}$ is between
$N^{\alpha} = N^{1}$ and $N^{\rho} = N^{2}$.

Fractional values

- α: independence number
- α^*: fractional independence number
  (max. weight of vertices s.t. each edge contains weight ≤ 1)
- ρ^*: fractional edge cover number
  (min. weight of edges s.t. each vertex receives weight ≥ 1)
- ρ: edge cover number

LP duality!



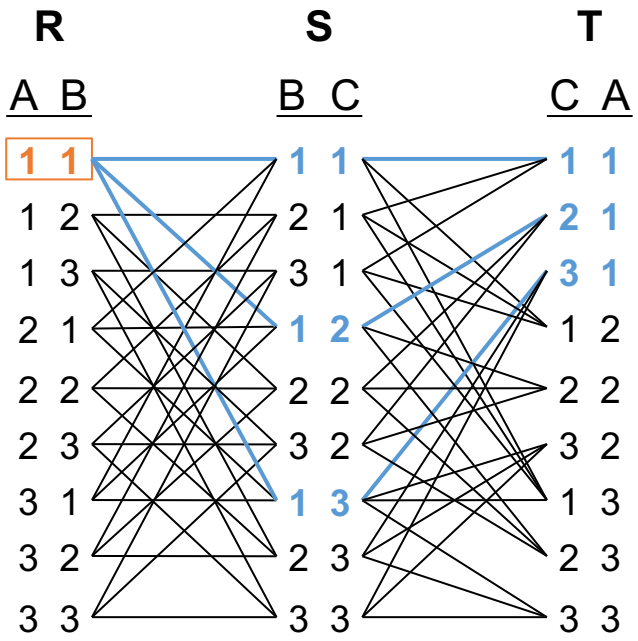
$\alpha = 1$ $\alpha^* = 3/2$ $\rho^* = 3/2$ $\rho = 2$

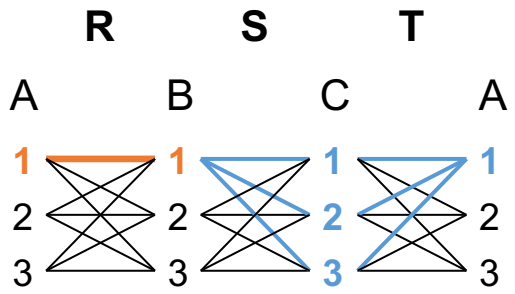
Example by: Marx. "Graphs, hypergraphs, and the complexity of conjunctive database queries", ICDT 2017. http://edbticdt2017.unive.it/marx-icdt2017-talk.pdf
Wolfgang Gatterbauer. Principles of scalable data management: https://northeastern-datalab.github.io/cs7240/

A tight example for AGM bound $O(n^{1.5})$ for triangle Q_Δ

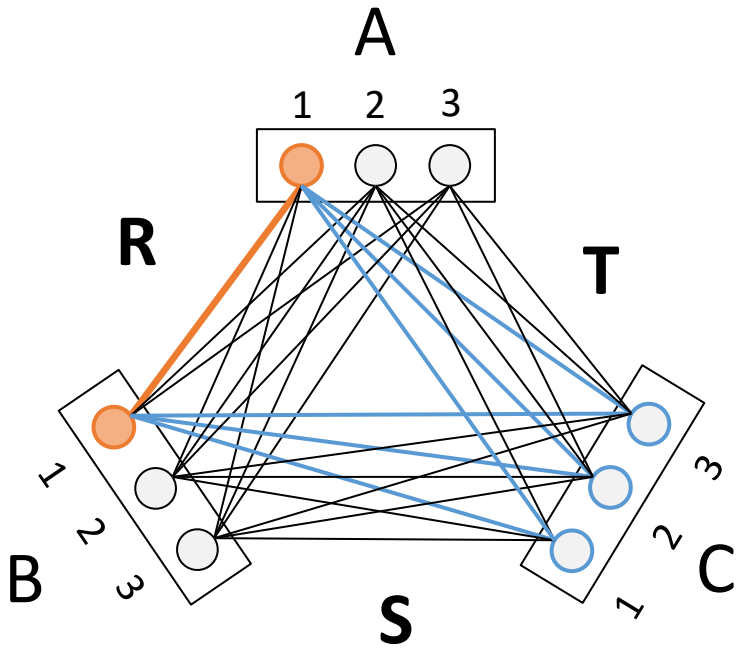
$Q_\Delta(A,B,C) = R(A,B) \bowtie S(B,C) \bowtie T(C,A)$

Q(x,y,z) :- R(x,y), S(y,z), T(z,x).





Notice every tuple is part of 3 join results, e.g. shown here for R(1,1)



$n = 9$ number of tuples per relation (= DB size for self-joins)

$m = \sqrt{n} = 3$ domain size

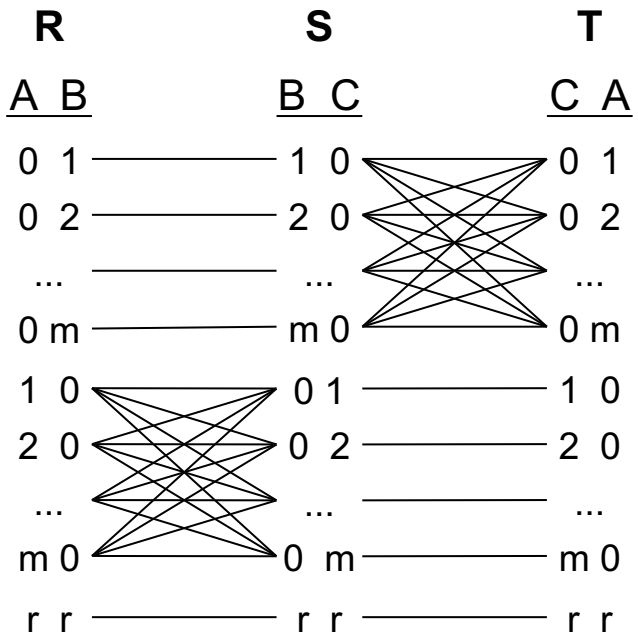
$|OUT| = n^{1.5} = 27$ output tuples

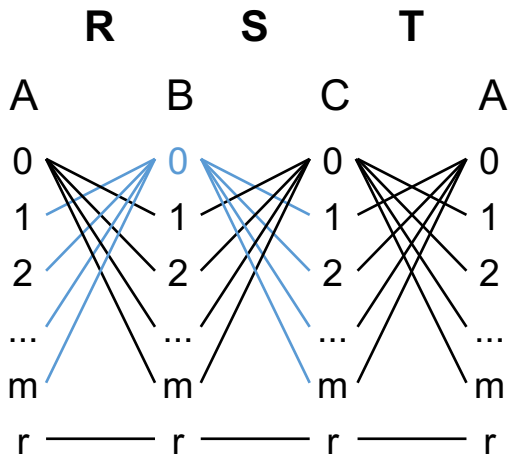
Q(x,y,z) :- R(x,y), R(y,z), R(z,x).

When binary joins give $O(n^2)$ intermediate sizes for Q_Δ

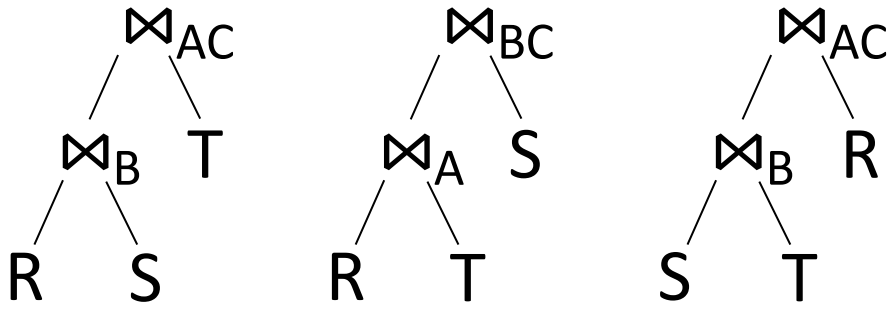
$Q_\Delta(A,B,C) = R(A,B) \bowtie S(B,C) \bowtie T(C,A)$

Q(x,y,z) :- R(x,y), S(y,z), T(z,x).





In whatever sequence we join the three tables, the size of the first join will always be $\Theta(n^2)$



$n = 2m + 1$ tuples per relation

$m + 2$ domain size

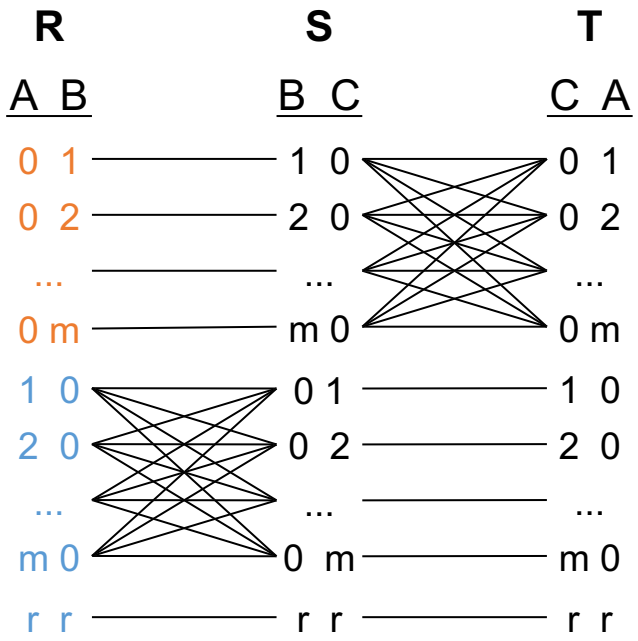
$|OUT| = 1$ output tuple

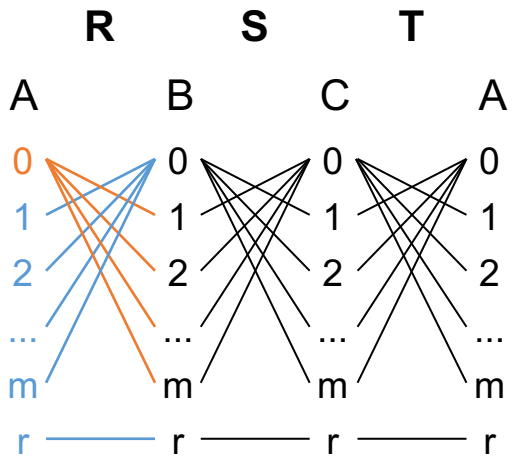
$|R \bowtie_B S| = |R \bowtie_A T| = |S \bowtie_B T| = m^2 = \Theta(n^2)$

Solution: partition the data

$Q_\Delta(A,B,C) = R(A,B) \bowtie S(B,C) \bowtie T(C,A)$

Q(x,y,z) :- R(x,y), S(y,z), T(z,x).



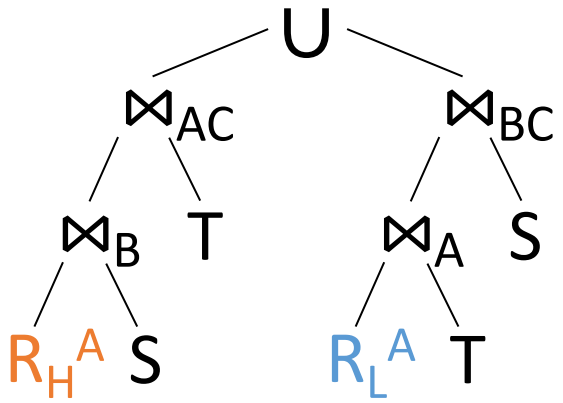


Trick: partition by outdegree, and use two plans in parallel!

$R = R_H^A \cup R_L^A$

$R_H^A = \{(a,b) \in R : |\sigma_{A=a}R| > n^{0.5}\}$

$R_L^A = \{(a,b) \in R : |\sigma_{A=a}R| \leq n^{0.5}\}$



$n = 2m + 1$ tuples per relation

$m + 2$ domain size

$|OUT| = 1$ output tuple

Algorithmica (1997) 17: 209–223

Algorithmica

Finding and Counting Given Length Cycles[1]

N. Alon,[2] R. Yuster,[2] and U. Zwick[2]

Abstract. We present an assortment of methods for finding and counting simple cycles of a given length in directed and undirected graphs. Most of the bounds obtained depend solely on the number of edges in the graph in question, and not on the number of vertices. The bounds obtained improve upon various previously known results.

k=2 for triangle

THEOREM 3.4. *Deciding whether a directed or undirected graph $G = (V, E)$ contains simple cycles of length exactly $2k - 1$ and of length exactly $2k$, and finding such cycles if it does, can be done in $O(E^{2-1/k})$ time.*

PROOF. We describe an $O(E^{2-1/k})$-time algorithm for finding a C_{2k} in a directed graph $G = (V, E)$. The details of all the other cases are similar. Let $\Delta = E^{1/k}$. A vertex in G whose degree is at least Δ is said to be of *high degree*. The graph $G = (V, E)$ contains at most $2E/\Delta = O(E^{1-1/k})$ high-degree vertices. We check, using Monien's

= "heavy": $\Delta = E^{1/2}$ for triangle

Source: Alon, Yuster, Zwick. "Finding and counting given length cycles", Algorithmica, 1997. https://doi.org/10.1007/BF02523189