Lecture #20

CS 170

Spring 2021

More NP-Complete Problems

- Review definitions of P, NP, etc
- All of NP → CSAT → SAT → 3SAT
- 3SAT → Independent Set (IS)

  IS ↙ Vertex Cover (VC)

  IS ↓ Clique

- 3SAT

  ↓

  3D Matching (3DM)

  ↓

  Zero-One Equations (ZOE)

  ZOE ↙ Integer Linear Prog (ILP)

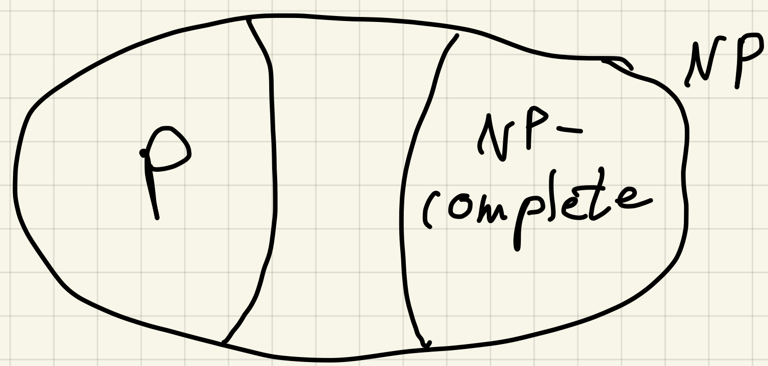
  ZOE ↘ Rudrata/Hamiltonian Cycle (RHC)

  ↓

  Traveling Salesperson Problem (TSP)

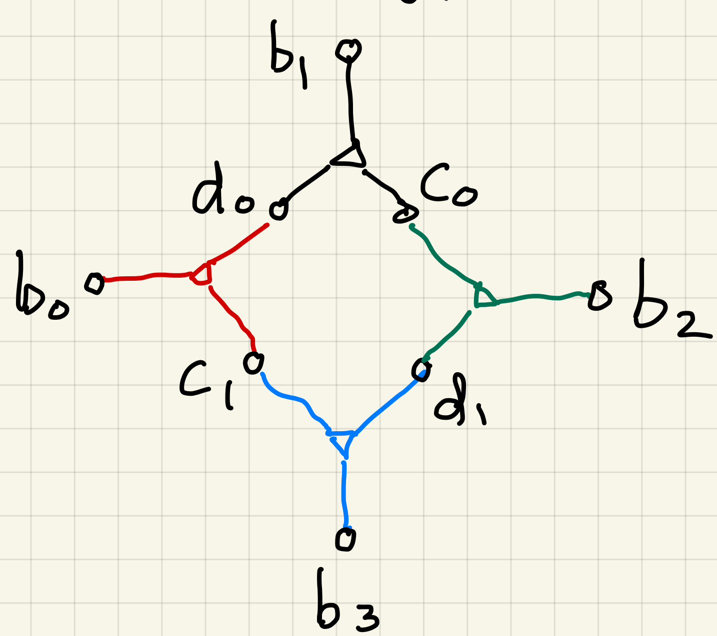
Defining NP-hard and NP-complete

- P = "complexity class" of all relations R such that decide(R) costs poly($|x|$) (P = "polynomial")
- NP = all relations R such that given x, $\exists\, w$ of size $|w| = \text{poly}(|x|)$, so $V_R(x, w)$ costs poly($|x|$) when $R(x,w)=1$ for some w
  - Ex: if $V_R(x,w)$ costs poly($|x|$)
- Def: problem A is NP-hard if $B \to A$ for all $B \in NP$
- Def: problem A is NP-complete if A NP-hard and in NP
  - NP-complete problems exist!



3SAT → 3D Matching (3DM) (1/3)

- 3DM: Given Sets $\{d_0, \dots, d_k\}$, $\{c_0, \dots, c_k\}$, $\{b_0, \dots, b_k\}$ and triples $\{(d_3, c_2, b_7), (d_1, c_3, b_2), \dots\}$: Is there a subset of triples where each d_i, c_i and b_i appears once?
- Need "gadgets" built from triples to model variables (T or F) and clauses $(x \vee \bar{y} \vee z)$
- Variable x: use 4 triples: $\triangle = (d_0, c_0, b_1)$, $\triangle$, $\triangle$, $\triangle$



to match all d_i, c_i need to pick either

$\triangle$ and $\triangle$: $x = T$

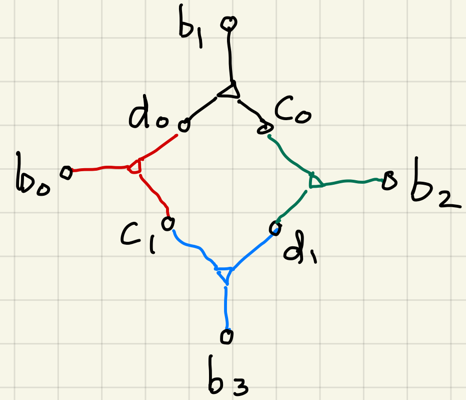
or

$\triangle$ and $\triangle$: $x = F$

3SAT → 3D Matching (3DM) (2/3)

- For each clause, $(x \vee \bar{y} \vee z)$: add d_c and c_c

x : add (d_c, c_c, b_{1x})
or (d_c, c_c, b_{3x})
so we need
to pick $\triangle, \triangle$
to make $x = T$



var$=T \Rightarrow \triangle, \triangle$

var $= F \Rightarrow \triangle, \triangle$

$\bar{y}$: add (d_c, c_c, b_{0y})
or (d_c, c_c, b_{2y})
so we need
to pick $\triangle, \triangle$
to make $y = F$

ditto for z

Assuming each literal $(x, \bar{x}, y, \bar{y}, \ldots)$ appears twice, they can all be connected, all b_i in one (d_c, c_c, b_i)

$\Rightarrow$ 3DM sets each var$=T$ or F to make each clause T

3SAT → 3D Matching (3DM) (3/3)

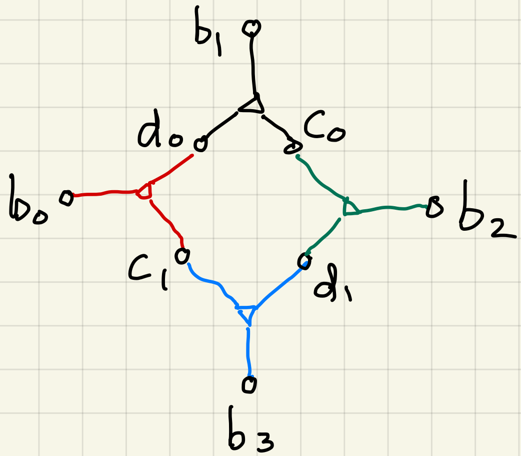
- What if each literal does not appear twice?
  - Suppose variable x appears $k \geq 3$ times (could be x or $\bar{x}$)
  - Replace each appearance by new variable x_k
  - Need to ensure all x_k are equal: add $(\bar{x}_1 \vee x_2) \wedge (\bar{x}_2 \vee x_3) \wedge \cdots \wedge (\bar{x}_{k-1} \vee x_k) \wedge (\bar{x}_k \vee x_1)$
  - Each literal appears at most twice
- What if some literal appears < twice?
  - Not enough triples to cover all b_i
  - If m triples missing, add

$$(\tilde{d}_i, \tilde{c}_i, b) \text{ for } i = 1 \text{ to } m, \text{ for all } b$$

to match left-over b

3D Matching (3DM) $\to$ Zero-one Equations (ZOE)

- ZOE: Solve (if possible) $Ax = 1$, each $A_{ij}, x_j \in \{0,1\}$
- A has
  - one column per triple
  - one row per d_i, c_i, b_i
  - $A_{ij} = 1$ if label of row i contained in label of column j



$$A = \begin{array}{c} \\ d_0 \\ d_1 \\ c_0 \\ c_1 \\ b_0 \\ b_1 \\ b_2 \\ b_3 \end{array} \begin{array}{c} \begin{array}{cccc} \triangle & \triangle & \triangle & \triangle \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

$x_j = 1$ means select triple labelling column j

- $(Ax)_i = \#$ selected triples containing label of row i
- Ex: $\left(A \cdot \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}\right)_1 = 2$ because d_0 in $\triangle$ and $\triangle$
- $Ax = 1$ iff each row label in one selected triple $\Leftrightarrow$ 3DM

ZOE (Zero-One Equations) $\to$ ILP (Integer LP)

- ILP: need to find a "feasible" x : $Ax \le b$
- Convert $Ax = 1$, $x_i \in \{0,1\}$, to inequalities:

$$Ax = 1 \longrightarrow Ax \le 1 \ , \ (-A)x \le -1$$

$$x_i \in \{0,1\} \longrightarrow x \le 1 \ , \ -x \le 0$$

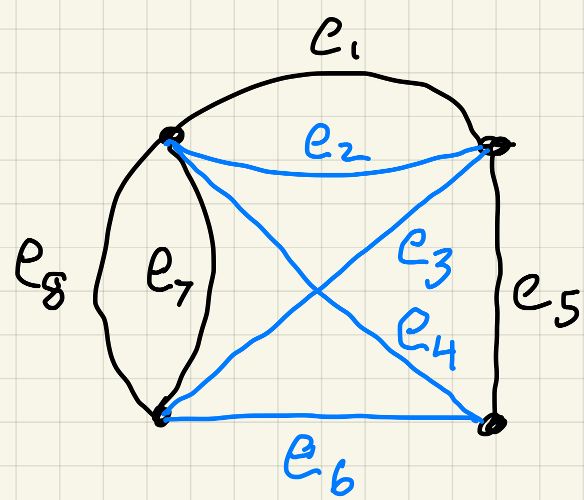
ZOE (Zero-One-Equations) (1/3)
$\longrightarrow$ RHC (Rudrata-Hamiltonian Cycle)

- RHC - find a cycle in G that visits each vertex once
- 2 Step Reduction:

  ZOE $\rightarrow$ RHC with paired edges (RHCwpe) $\rightarrow$ RHC

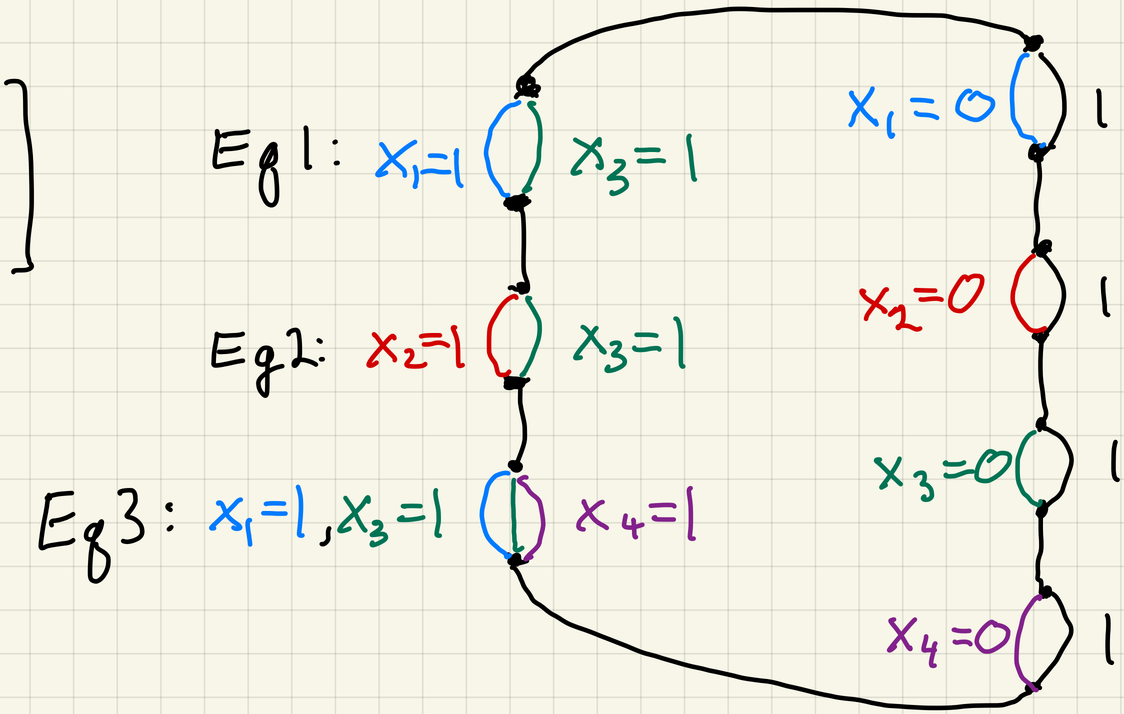
- RHCwpe: Given G and a set of edge pairs $C = \{(e_i, e_i')\}$ find a cycle that visits each vertex once, and uses either e_i or e_i', for each pair $(e_i, e_i') \in C$

Ex: $C = \{(e_1, e_3), (e_5, e_6), (e_4, e_5), (e_3, e_7), (e_3, e_8)\}$



ZOE → RHC with paired edges (RHCwpe) (2/3)

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$



- A cycle "chooses" value of each x_i on right side and one nonzero x_i per equation on left side
- Enforce consistent choices on left and right:

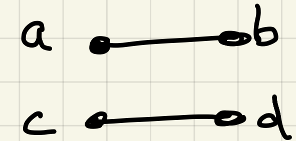
$$C = \{ (\underset{\text{Eq1}}{x_1=1}, x_1=0), (\underset{\text{Eq3}}{x_1=1}, x_1=0), (\underset{\text{Eq2}}{x_2=1}, x_2=0), (\underset{\text{Eq1}}{x_3=1}, x_3=0), \dots \}$$

- Can solve $Ax=1$ iff can find RHC with paired edges

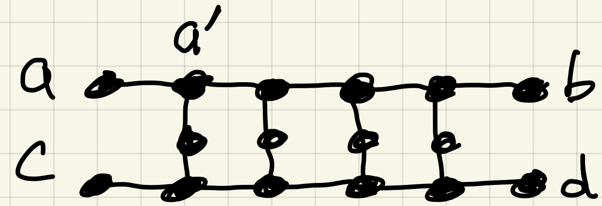
RHC with paired edges (RHCwpe) → RHC (3/3)

- Need to enhance G to enforce choices in C

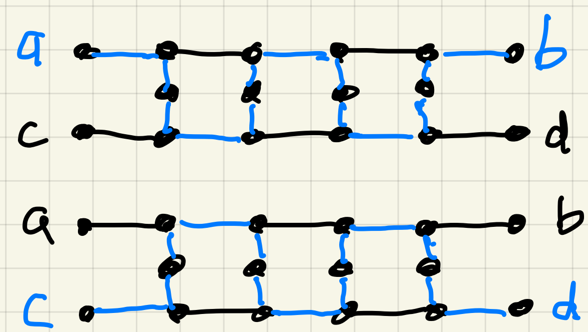
- Gadget: $(e, e') = ((a,b),(c,d))$



add vertices and edges:



Only 2 paths possible to touch all new vertices once:



need to choose (a, b) or (c, d)

- If (a,b) appears more than once in C: replace additional (a,b) by (a, a'), repeat

Rudrata-Hamiltonian Cycle (RHC)

$\rightarrow$ Traveling Salesperson Problem (TSP)

- RHC – find cycle visting each vertex once
- TSP – find shortest cycle visting each vertex once
- Given input $G(V,E)$ for RHC, create new graph G'
    - G' has same vertices V as G
    - Add each edge in E to G' with weight $=1$
    - Add all other edges not in E to G' with weight $1+\alpha$, $\alpha > 0$
- # edges in a cycle visiting each vertex once $= |V|$
- Total weight of shortest cycle visiting each vertex once $= |V|$ iff it only uses edges in E, else $\geq |V| + \alpha$
- TSP finds shortest cycle of weight $|V|$ iff same cycle solves RHC

$\parallel$