Exercises: ① INDEP SET $\le^P_m$ CLIQUE

② 3-COLORING $\le^P_m$ SAT

Formalization of "search problems":

Df: Class of languages NP:

$L \in NP \equiv \exists V \in P$ (verifier)

∃ certificate — of polynomial size — which is accepted by the verifier

$\forall \alpha \in \Sigma^*: \alpha \in L \Leftrightarrow \left(\exists \beta \in \Sigma^*: |\beta| \in poly(|\alpha|) \ \&\ V(\alpha, \beta) \right)$

$P \subseteq NP$ ... verifier does all the work & ignores β

resembles proofs in logic: true statements have a proof, which is easy to verify
for false statements, no proof passes verification

Big question: Is $P = NP$?

1 M\$ price by Clay Mathematical Institute (waits for you ☺)

Df: Language L is NP-hard $\equiv \forall K \in NP: K \le^P_m L$

L is NP-complete $\equiv$ furthermore, $L \in NP$

Lemma: Let $K \le^P_m L$. Then:

① if $L \in NP$, then $K \in NP$ (just compose verifier with reduction)

② if K is NP-hard, then L is NP-hard. ($\forall M \in NP \ M \le K \le L \Rightarrow M \le L$)

②' if K is NP-complete & $L \in NP$, then L is NP-complete.

makes it easy to prove NP-completeness once we have one NP-comp. problem

Lemma: If $L \in P$ is NP-complete, then $P = NP$.

Proof: $P \subseteq NP$ is trivial, will prove $NP \subseteq P$:
Let $K \in NP$. Then $K \le L$, which implies $K \in P$.

Thm (Cook-Levin): SAT is NP-complete.
↳ will be proven later

MORE REDUCTIONS

3D MATCHING Input: sets B (boys), G (girls), C (cats)
$J \subseteq B \times G \times C$ (triples)
Output: $\exists J' \subseteq J$ s.t. each element of $B \cup G \cup C$ is contained in exactly 1 triple in J'

(generalizes bipartite matching, which is in P)

3-SAT: SAT, but all clauses contain at most 3 literals (generally: k-SAT)

3,3-SAT: Furthermore, every variable occurs in at most 3 clauses. (generally: k,l-SAT)
[Extension: every literal occurs at most 2 times – i.e., the 3 occurs of a variable aren't all positive nor all negative.]

Reduction: SAT $\leq_m^P$ 3-SAT

$$\underbrace{(l_1 \vee l_2 \vee \_\_ \vee l_k)}_{\text{"long" clause with } k>3 \text{ literals}} \xrightarrow[\text{by}]{\text{replace}} \overbrace{(l_1 \vee l_2 \vee t)}^{3 \text{ literals}} \;\&\; \overbrace{(l_3 \vee \_\_ \vee l_k \vee \neg t)}^{k-1 \text{ literals}}$$

t: new variable

New formula is satisfiable $\Leftrightarrow$ the old one was.
Iterate until all long clauses are broken.

Reduction: 3-SAT $\leq_m^P$ 3,3-SAT

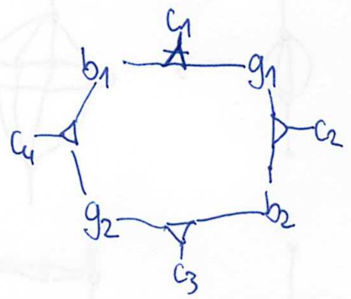
Replace variable x with $k>3$ occurrences by new variables x_1 — x_k.
Add clauses $(x_1 \Rightarrow x_2), (x_2 \Rightarrow x_3), \dots, (x_{k-1} \Rightarrow x_k), (x_k \Rightarrow x_1)$

this is $\neg x_2 \vee x_3$

Preserves satisfiability.
Each new variable has at most 2 positive & at most 2 negative occurrences.
Can apply the transform for $k=3$, too.

Reduction: 3,3-SAT $\leq_m^P$ 3D-MATCHING

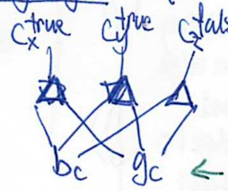
Choice gadget (for each variable)



2 states: c_1, c_3 free ← logical 0
and c_2, c_4 free ← logical 1

b_1, b_2, g_1, g_2 unique for this gadget
$c_1 \dots c_4$ shared with clause gadgets

(also called consistency gadget)
Clause gadget for $x \vee y \vee \neg z$



← unique for this gadget

1 of the cats which are free if x is true (that is c_2 or c_4)

← each literal occurs at most 2 times in 3,3-SAT formulas, so we have enough cats for all literals

we have (4 · #variables – Σ of clause sizes) free cats $\Rightarrow$ add this many pairs of "universal cat lovers" which have triples for every cat

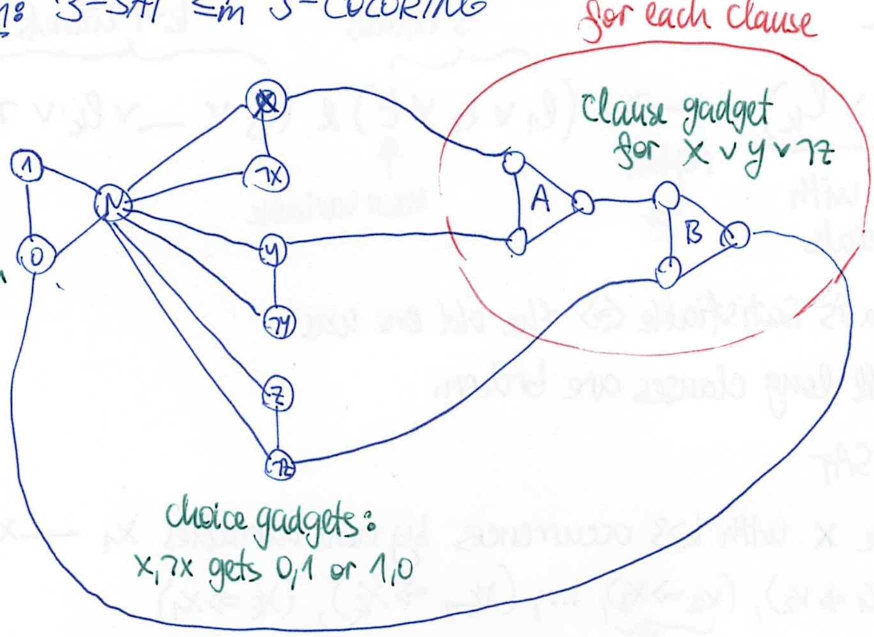
$\exists$ matching $\Leftrightarrow$ $\exists$ satisfying assignment

Exercise: 3D-MATCHING $\leq^P_m$ ZOE (zero-one equations)

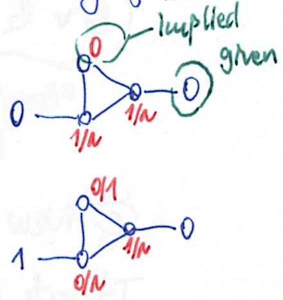
↳ & show that restriction of ZOE to equations with exactly 3 variables stays NP-complete.
[This is sometimes called 1-in-3-SAT: exactly 1 literal must be true ... no negations are needed. There also exists a direct reduction from 3-SAT to this problem.]

Reduction: 3-SAT $\leq^P_m$ 3-COLORING

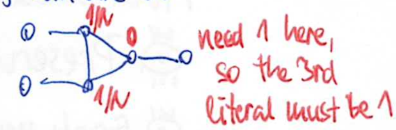
vertices of this Δ get 3 different colors, let's label them 0, 1, N



= coloring of B:



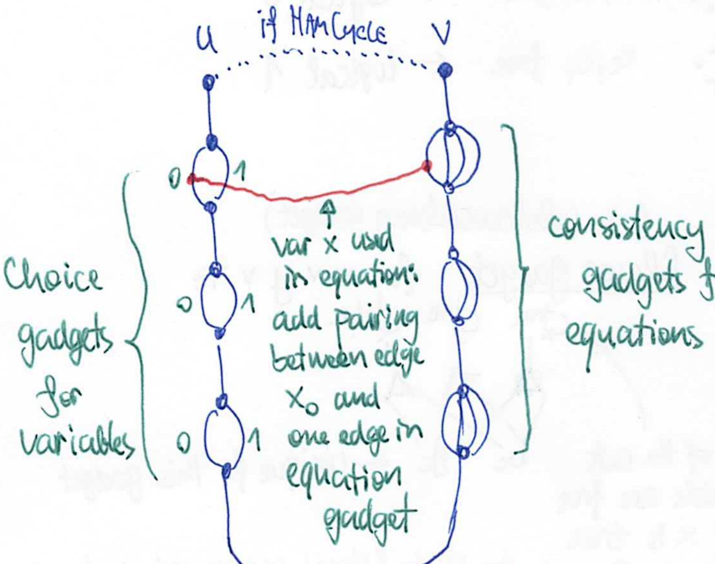
= coloring of A:
- if inputs contain 1: use coloring of B which passes 0
- if both are 0:

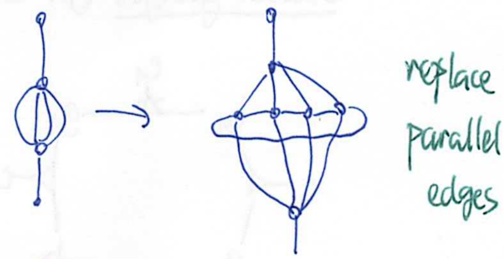


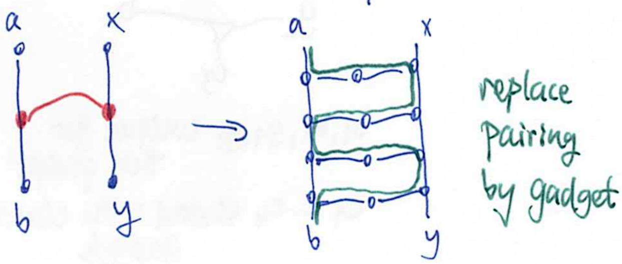
Reduction: ZOE $\leq^P_m$ HAMILTON ~~CYCLE~~ or PATH

First consider problem HAMPATH* which allows:
- parallel edges
- pairing: e ~ f ≡ must use exactly 1 edge of e, f

reduce to plain HAMILTON PATH







Exercises:
- SUBSET SUM problem is NP-complete: Given a finite set $X \subseteq \mathbb{N}$, $s \in \mathbb{N}$, is there $X' \subseteq X$ s.t. $\sum_{a \in X'} a = s$? (Hint: reduce from ZOE)
- 2 BANDITS: given finite $X \subseteq \mathbb{N}$, is there $X' \subseteq X$ s.t. $\sum X' = \sum (X \setminus X')$? Also NP-complete.