Logic and theory of computation

theory of computation part, 5th lecture

NP-completeness

Reminder:

NP-complete language

L is **NP-complete** (for polynomial time reduction), if

- $L \in$ NP
- L NP-hard, i.e., for all $L' \in$NP $L' \leq_p L$ holds.

So NP-complete problems (if there are any) are the hardest problems of NP. No polynomial algorithm is known for them and it is not probable that there will be any. This is due to our previous observation that a polynomial time algorithm for an NP-complete problem would mean P=NP.

Is there any NP-complete problems at all?

SAT= $\{\langle\varphi\rangle \mid \varphi$ is a satisfiable propositional formula in CNF$\}$

Theorem (Cook)

SAT is NP-complete.

Proof of Cook's Theorem

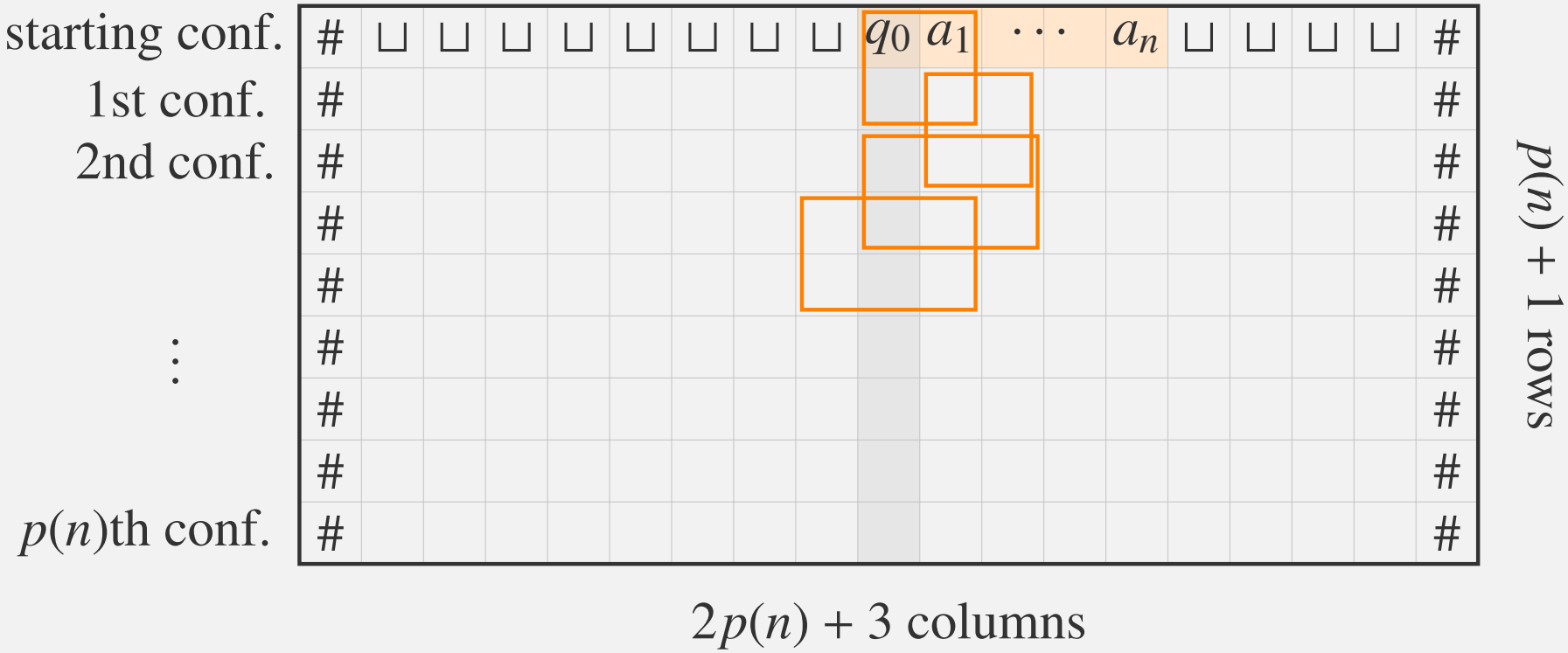
Proof:

- SAT$\in$ NP: Given φ as an input. A NTM produces an interpretation I in polynomial time. Then checks in polynomial time whether $I \models \varphi$-t.
- SAT is NP-hard. For this let $L \in$ NP be arbitrary, we need $L \leq_p$ SAT.

  • Let $M = \langle Q, \Sigma, \Gamma, \delta, q_0, q_a, q_r \rangle$ be a NTM deciding L in $p(n)$ polynomial time (Assume $p(n) \geq n$.)
  • Let $w = a_1 \cdots a_n \in \Sigma^*$ be a word.
  • We construct a CNF φ_w in polynomial time satisfying $w \in L \Leftrightarrow \langle \varphi_w \rangle \in$ SAT.
  • a computation of M on w can be described by a table T where
  – the 1st row is $\# \sqcup^{p(n)} C_0 \sqcup^{p(n)-n} \#$, where $C_0 = q_0 w$ is the starting configuration of M for w
  – Consecutive rows of T are consecutive configurations of M (extended by enough $\sqcup$'s and #'s at the begining and at the end). The rows are of length $2p(n) + 3$.

Proof of Cook's Theorem (cont'd.)

– there are $p(n) + 1$ rows. An accepting configuration is repeated till the last row if it is reached earlier.



– by the definition of yields in one step the difference of two consecutive rows can be cowered by a 2x3 "window "

– height of T is big enough to contain all computations of length $\leq p(n)$. The number of ⊔'s ($\Rightarrow$ width of T) is set to ensure these "windows" do not "fall off" at the sides.

Proof of Cook's Theorem (cont'd)

– Atoms of φ_w are of the form $p_{i,j,s}$, where: $T(i, j) = s$, for some $s \in \Delta = Q \cup \Gamma \cup \{\#\}$.

– φ_w describes all possible computations of M for $w$of length $\leq p(n)$. φ_w is of the form $\varphi_w = \varphi_0 \wedge \varphi_{\text{start}} \wedge \varphi_{\text{move}} \wedge \varphi_{\text{accept}}$.

– φ_0 is true if and only iff all cells contain exactly one symbol

$$\varphi_0 := \bigwedge_{\substack{1 \leq i \leq p(n)+1 \\ 1 \leq j \leq 2p(n)+3}} \left(\left(\bigvee_{s \in \Delta} p_{i,j,s} \right) \wedge \bigwedge_{s,t \in \Delta, s \neq t} \left(\neg p_{i,j,s} \vee \neg p_{i,j,t} \right) \right)$$

– φ_{start} is true if and only iff the first row of T is the starting configuration extended by enough $\sqcup$'s and #'s.

$$\varphi_{\text{start}} := p_{1,1,\#} \wedge p_{1,2,\sqcup} \wedge \cdots \wedge p_{1,2p(n)+2,\sqcup} \wedge p_{1,2p(n)+3,\#}$$

Proof of Cook's Theorem (cont'd.)

– φ_{move} is true if all "windows" are legal according to δ:

$$\varphi_{\text{move}} := \bigwedge_{\substack{1 \le i \le p(n) \\ 2 \le j \le 2p(n)+2}} \psi_{i,j},$$

where $\psi_{i,j} \sim \bigvee_{\substack{(b_1,\ldots,b_6) \\ \text{legal window}}} p_{i,j-1,b_1} \wedge p_{i,j,b_2} \wedge p_{i,j+1,b_3} \wedge p_{i+1,j-1,b_4} \wedge p_{i+1,j,b_5} \wedge p_{i+1,j+1,b_6}$

b_1	b_2	b_3
b_4	b_5	b_6

But: $\psi_{i,j}$ is not a disjunction, change it to

$$\psi_{i,j} := \bigwedge_{\substack{(b_1,\ldots,b_6) \\ \text{illegal window}}} \neg p_{i,j-1,b_1} \vee \neg p_{i,j,b_2} \vee \neg p_{i,j+1,b_3} \vee \neg p_{i+1,j-1,b_4} \vee \neg p_{i+1,j,b_5} \vee \neg p_{i+1,j+1,b_6}$$

Proof of Cook's Theorem (cont'd.)

– finally: φ_{accept} is true if and only if there is a q_a in the last row

$$\varphi_{\text{accept}} = \bigvee_{j=2}^{2p(n)+2} p_{p(n)+1,j,q_a}$$

.

• $w \in L \Leftrightarrow$ NTM M has a computation accepting $w \Leftrightarrow T$ can be filled so that ϕ_w is true $\Leftrightarrow$ ϕ_w is satisfiable $\Leftrightarrow \langle\varphi_w\rangle \in$ SAT,

• how many literals are there in φ_w? Let $k := |\Delta|$.

ϕ_0 : $(p(n)+1)(2p(n)+3)(k+k(k-1)) = O(p^2(n))$,

φ_{start} : $2p(n)+3 = O(p(n))$,

φ_{move} : $\leq p(n)(2p(n)+1)k^6 \cdot 6 = O(p^2(n))$,

φ_{accept} : $2p(n)+1 = O(p(n))$,

so φ_w has size $O(p^2(n))$, that can be constructed in polynomial time

• so $w \mapsto \langle\varphi_w\rangle$ is a pol. time reduction, i.e., $L \leq_p$ SAT.

• This holds for all $L \in$NP. So SAT is NP-hard. Being in NP means it is NP-complete.

Further NP-complete problems, kSAT

Theorem

If L is NP-complete, $L \leq_p L'$ and $L' \in$ NP, then L' is NP-complete as well.

Proof: Let $L'' \in$ NP arbitrary. Since L is NP-complete $L'' \leq_p L$ holds. By the conditions $L \leq_p L'$, so by the transitivity of polynomial time reductions ($p_1(p_2(n))$ is a polynomial as well) L' is NP-hard. The statement follows from this and the 3rd condition.

So polynomial time reduction can be used to prove NP-completeness of other languages.

kSAT=$\{\langle\varphi\rangle \mid \varphi$ is a satisfiable CNF formula in propositional logic with each term having exactly k literals$\}$.

Such formulas are called kCNF's (CNF with exactly k literals/term) in the future.

3SAT NP-completeness

Theorem

3SAT is NP-complete.

- 3SAT is in NP, see SAT
- SAT$\leq_p$ 3SAT
  We need $f : \varphi \mapsto \varphi'$, φ CNF, φ' 3CNF, φ' is satisfiable $\Leftrightarrow$ φ satisfiable, f is a polynomial time reduction.

$\varphi \mapsto \varphi'$:

ℓ	$\ell \vee q \vee r,\ \ell \vee q \vee \neg r,\ \ell \vee \neg q \vee r,\ \ell \vee \neg q \vee \neg r$
$\ell_1 \vee \ell_2$	$\ell_1 \vee \ell_2 \vee q,\ \ \ell_1 \vee \ell_2 \vee \neg q$
$\ell_1 \vee \ell_2 \vee \ell_3$	$\ell_1 \vee \ell_2 \vee \ell_3$
$\ell_1 \vee \ell_2 \vee \ell_3 \vee \ell_4$	$\ell_1 \vee \ell_2 \vee X,\ \ \neg X \vee \ell_3 \vee \ell_4$
$\ell_1 \vee \cdots \vee \ell_n$ $(n \geq 5)$	$\ell_1 \vee \ell_2 \vee p_1, \neg p_1 \vee \ell_3 \vee p_2, \ldots, \neg p_{n-2} \vee \ell_{n-1} \vee \ell_n$

$q, r, p_1, \ldots, p_{n-2}$ are new atoms. φ' is the conjunction of these.

Remark: HORNSAT: see SAT, but max. 1 positive literal/term.
HORNSAT and 2SAT $\in$ P.

3 colorabilty of a graph

A graph is *k*-**colorable**, if its nodes can be colored by k colors, so that there are no two adjacent nodes with the same color.

3COLOR=$\{\langle G\rangle \mid G$ 3-colorable$\}$

Theorem

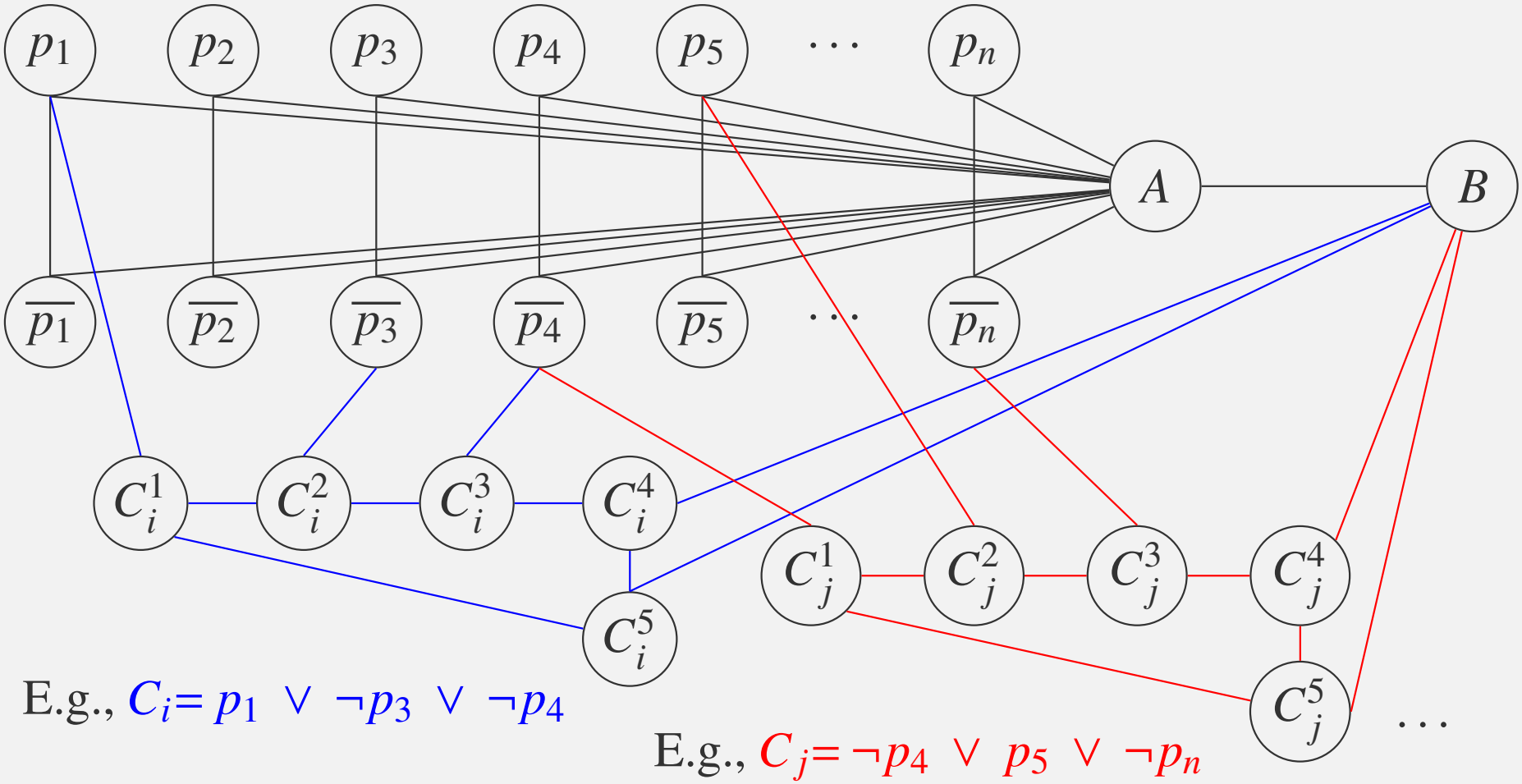
3COLOR is NP-complete.

- 3COLOR is in NP: let a branch of a NTM correspond to a coloring. Checking soundness of a coloring can be done in polynomial time.
- 3SAT$\leq_p$ 3COLOR

  Enough to construct a graph G_φ for a 3CNF formula φ in polynomial time such that φ is satisfiable $\Leftrightarrow$ G_φ 3 colorable.

3 colorability

Let $p_1, \ldots, p_n$ be the atoms appearing in φ. Furthermore let $\varphi = C_1 \wedge \ldots \wedge C_m$, where $C_1, \ldots C_m$ φ are clauses of exactly 3 different literals. Construction of G_φ:



E.g., $C_i = p_1 \vee \neg p_3 \vee \neg p_4$

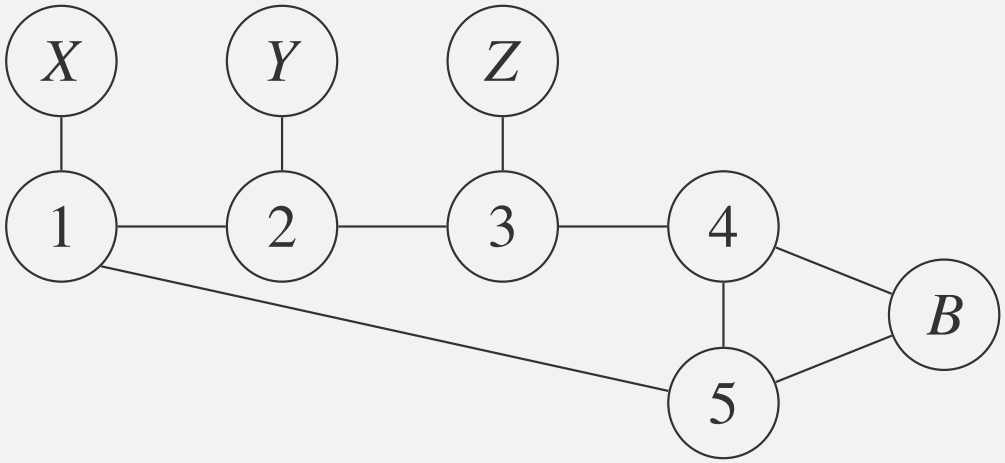
E.g., $C_j = \neg p_4 \vee p_5 \vee \neg p_n$

A pentagon corresponds to each of the clauses as on the figure.

3 colorability

Lemma: Let G_0 be a graph, as below, where X, Y, Z, B are colored with 2 colors. There exists an extension of this partial coloring for G_0 if and only if X, Y, Z, B are not monochromatic.

G_0



Proof of the lemma:

- If X, Y, Z, B are monochromatic, there's no extension, since a pentagon can not be colored by 2 colors.
- Otherwise consider the following coloring.
  1st step: we use only 2 colors, we color 1,2,3,4,5 to the opposite colorof its neighbor from $\{X, Y, Z, B\}$.

3 colorability

This coloring is not good yet, as among 1,2,3,4,5 there can be neighbors with the same color.

2nd step: use the 3rd color. If there is cyclically consecutive set of 1,2,3,4,5 with the same color, then color every second node clockwise for the 3rd color.

Proof of reducibility:

- Suppose that φ is satisfiable, let's color G_φ with 3 colors. Let the colors be red, green and blue. If p_i is true, then let the node p_i be green and $\overline{p_i}$ be red. If it is false, let it be the opposite. Let A be blue and B red. Since all clauses are true all the pentagons have both a green (the true literal) and a red neighbor (B), so this coloring can be extended for all pentagons.

3 colorability

- Suppose that G_φ is colored by 3 colors. Wlog. let A be blue. $p_1, \ldots, p_n, \overline{p_1}, \ldots, \overline{p_n}$ are all neighbors of A so none of them can be blue. Furthermore the pairs $(p_i, \overline{p_1})$ are connected, so they have 1 green and 1 red color. Wlog. B is red (the green case is similar). Since all pentagons are colored there should be a green neighbor for all of them by the lemma. Interpretation "p_i:=true $\Leftrightarrow$ node p_i is "green" satisfies φ.

So $\varphi \mapsto G_\varphi$ is a reduction. As G_φ can be built from φ in polynomial time, this is a polynomial time reduction.

Since 3SAT is NP-complete these together imply the NP-completeness of 3COLOR.

Remark: 2COLOR $\in$ P

3 more graph problems

For these problems G is a simple, undirected graph and k is a netural number. A complete subgraph of G is called a **clique**, an empty subgraph is called an **independent set**.

CLIQUE=$\{\langle G, k\rangle \mid G$ has a clique of size $k\}$

INDEPENDENTSET= $\{\langle G, k\rangle \mid G$ has an independent set of size $k\}$

Let $S \subseteq V(G)$ and $E \in E(G)$. If $S \cap E \neq \emptyset$, then we say that S **covers** the edge E. If S covers all edges $E \in E(G)$, then S is a **vertex cover**.

VERTEXCOVER=$\{\langle G, k\rangle \mid G$ has a vertex cover of size $k\}$

If G has a clique/independent set of size k then it has one for all smaller k's. If it has a vertex cover of size k then it has one for all larger k's ($k \leq |V(G)|$).

IndependentSet

Theorem

Clique, IndependentSet, VertexCover are NP-complete.

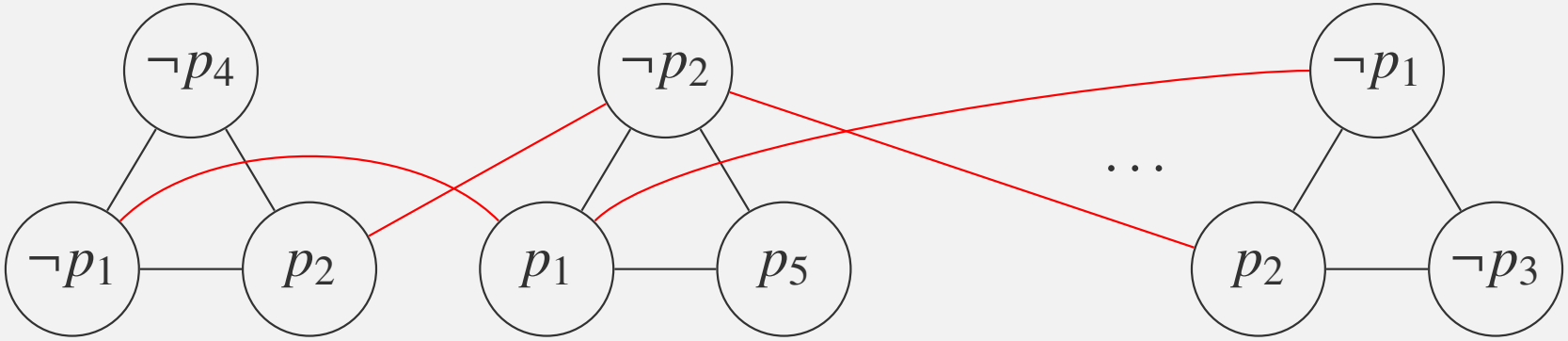
- A NTM can check a k-element subset on one of its branches. For all of hese 3 languages both constructing k-element subset and checking it are polynomial.
- 3SAT$\leq_p$ IndependentSet

  We need: $f : \varphi \mapsto (G_\varphi, k)$, φ is a 3CNF and G_φ has an independent set of size k if and only if φ is satisfiable.

  Construction of (G_φ, k): for all clauses $L_1 \vee L_2 \vee L_3$ add a triangle disjoint from the other triangles. Assign the literals to the nodes. If φ has m clauses G_φ has size $3m$. Additionally connect all complementary pairs by an edge. $k := m$.

IndependentSet

An example:



* If φ is satisfiable, then all clauses have a true literal, choose one from each clause, the corresponding vertices are an independent set of size m.

* If G_φ has m independent vertices, than it must contain exactly one per triangle. Consider one such set, there are no compementary pairs among these literals (they are connected), so this set corresponds to a partial interpretation which evaluates all the clauses for true. Complete this interpretation arbitrarily.

CLIQUE, VERTEXCOVER

- INDEPENDENTSET $\leq_p$ CLIQUE

  $f : (G, k) \mapsto (\bar{G}, k)$

  A clique in G is an independent set in $\bar{G}$ and vice versa.

- INDEPENDENTSET $\leq_p$ VERTEXCOVER

  $f : (G, k) \mapsto (G, |V(G)| - k)$

  If G has an independent set F of size k then there is vertex cover of size $|V(G)| - k$ (the complement of F).

  If G has a vertex cover L of size $|V(G)| - k$ then there is an independent set of size k (the complement of L).

Both reductions are of polynomial time.