13 Oct 2023

NP-Complete Problems in Graph Theory

To show a problem A is NP-Complete you need two poly-time algorithms

1. Verifier V: takes an input of A (x) and a proposed solution (y) and verifies y is a valid solution to x.

$$A(x) = 1 \iff \exists y \; V(x, y) = 1$$

E.g. for 3SAT $V(x,y)$ takes logical formula x truth assignment y, outputs 1 iff y satisfies x.

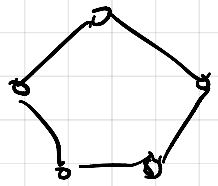
2. Reduction R: takes an input x_0 of some other problem known to be hard, e.g. 3SAT, and transforms x_0 into an input of A, x_1.

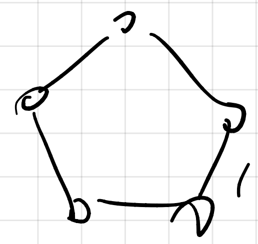
$$3SAT(x_0) = 1 \iff A(x_1) = 1.$$

__Ex.__ INDEPENDENT SET.

Input: undirected graph G, positive integer k.

Output: 1 if and only if $\exists$ a subset of k vertices of G, s.t. no edge has both of its endpoints in the subset.

E.g. , $2 \implies 1$ (yes there is a 2-element ind. set)



$, 3 \Rightarrow \emptyset$

Why NP-Complete?

① $V(x,y)$ takes $x = (G, k)$

and y = binary string encoding subset of $V(G)$, say S.

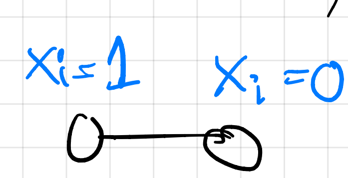
It checks $|S| = k$. $O(n)$

For each edge (u,v) it checks $u \notin S$ or $v \notin S$.

$O(m)$

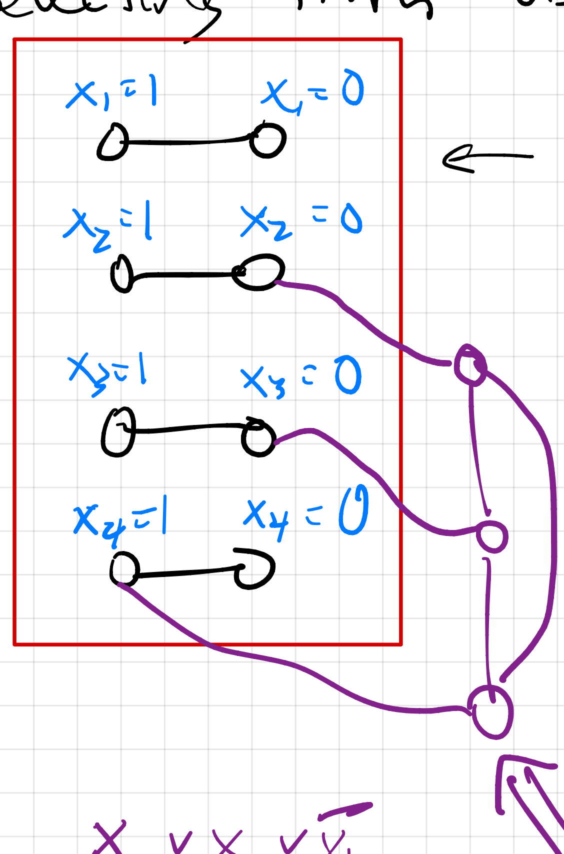
② We will reduce 3SAT to IND SET.

A structure that looks like a Boolean variable, translated to IND SET:



1-element independent subsets of this graph are in 1:1 correspondence with truth assignments of x_i.

Selecting truth assignments for n variables:



← n element independent subsets of this graph are in 1:1 corresp. with truth assignments of $x_1, \ldots, x_n$.

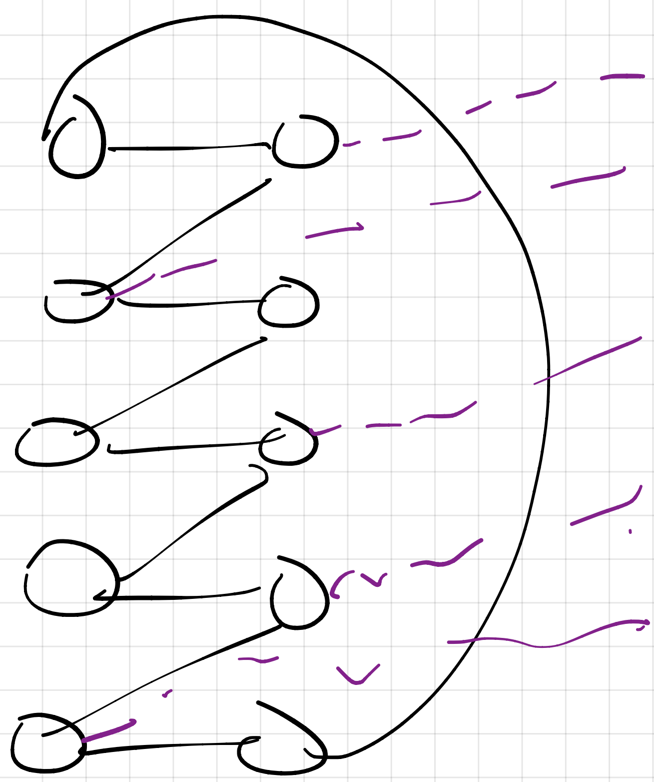
$x_2 \vee x_3 \vee \overline{x_4}$

⇑ Adjoin a gadget like this for every clause. Set $k = (\#\text{vars}) + (\#\text{clauses})$

IND SET RESTRICTED TO GRAPHS OF MAX DEGREE 3.

(d3-IND-SET).

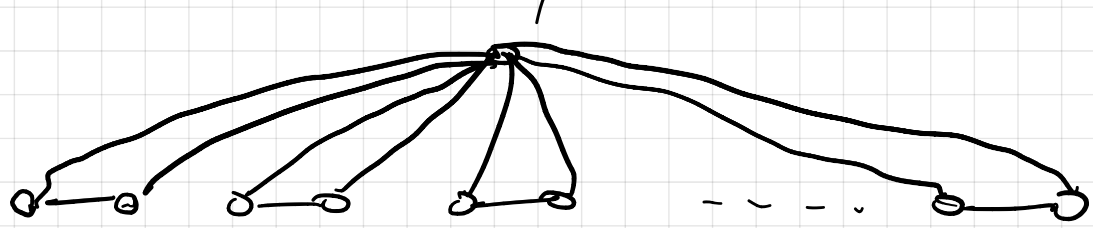
Gadget for a variable x_1 that belongs to 5 clauses

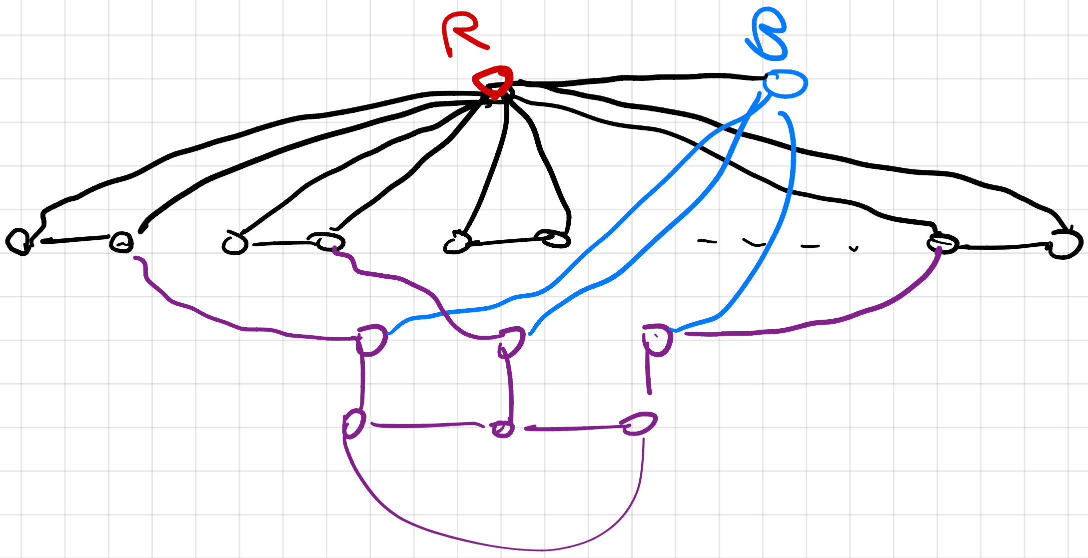


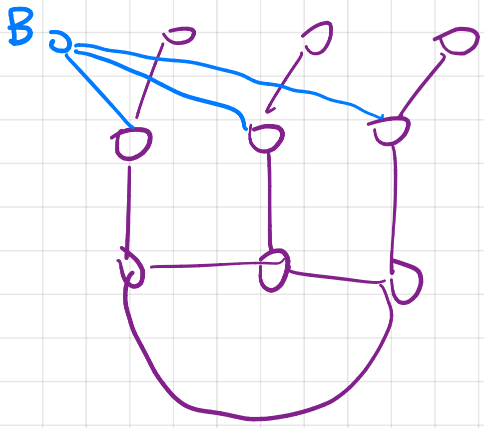
Make each variable x_i into a gadget with $2n_i$ vertices & edges forming an even cycle, where n_i denotes # clauses containing x_i or $\overline{x_i}$.

Set $k = \left(\sum_i n_i\right) + (\#\text{ clauses})$

GRAPH 3-COLORABILITY: Given undirected G, can we color its vertices with 3 colors such that the endpoints of every edge are differently colored.




R
B


B